

Universitat de Lleida
Escola Politècnica Superior
Enginyeria en Informàtica

Sistemes informàtics (Treball final de carrera)

**Redisseny del lloc web del grup Griho
amb el gestor de continguts OpenCms**

Autora: **Anna Cristina Comas i Aleix**
Director: **Toni Granollers Saltiveri**
Juliol 2007

Universitat de Lleida
Escola Politècnica Superior
Enginyeria en Informàtica

Sistemes informàtics (Treball final de carrera)

Redisseny del lloc web del grup Griho amb el gestor de continguts OpenCms

Director: **Toni Granollers Saltiveri**

Autora: **Anna Cristina Comas i Aleix**

Juliol 2007

*A tota la meva família,
pel seu suport en tot moment i
pels ànims que m'han donat.*

*Als companys de Griho,
gràcies per col·laborar en aquest projecte
fent-me gaudir del treball en equip.*

Resum

En aquest treball s'explica tot el procés seguit per al redisseny del lloc web del grup de recerca en Interacció Persona-Ordinador i Integració de bases de dades (GRIHO) de la Universitat de Lleida utilitzant OpenCms.

Per tal de portar a terme el projecte es segueix el model de procés definit en la tesi doctoral d'en Toni Granollers titulada "MPIu+a. Una metodología que integra la ingeniería del software, la integración persona-ordenador y la accesibilidad en el contexto de equipos de desarrollo multidisciplinares". Així, en el treball, trobarem explicades cadascuna de les etapes d'aquest model de procés juntament amb les tècniques que s'han aplicat en cadascuna per al desenvolupament de la nova web.

La plataforma tecnològica sobre la qual s'implementa el lloc web i en la qual s'allotjarà és el sistema gestor de continguts OpenCms, per tant, en aquest treball també hi podrem trobar informació útil sobre les característiques i funcionalitats que ofereix aquest sistema i una descripció acurada del procediment de creació d'un lloc web, detallant tots els elements que hi intervenen.

Índex de continguts

Capítol 1

| | |
|-----------------------------------|---|
| Introducció..... | 4 |
| 1.1. Introducció..... | 4 |
| 1.2. Objectius..... | 5 |
| 1.3. Temporalització..... | 6 |
| 1.4. Estructura del document..... | 7 |

Capítol 2

| | |
|-------------------------------------------------|----|
| Estat de l'art..... | 8 |
| 2.1. Tecnologies relacionades..... | 8 |
| 2.1.1. MPIu+a..... | 8 |
| 2.1.2. Sistemes gestors de continguts..... | 10 |
| 2.2. Estudis realitzats per fer el treball..... | 14 |

Capítol 3

| | |
|---------------------------------------------------------------|----|
| OpenCms..... | 15 |
| 3.1. Introducció..... | 15 |
| 3.1.1. Breu història d'OpenCms..... | 16 |
| 3.1.2. La comunitat OpenCms i alguns llocs de referència..... | 18 |
| 3.1.3. Descripció tècnica i components d'OpenCms..... | 20 |
| 3.2. Instal·lació d'OpenCms..... | 22 |
| 3.3. L'entorn de treball d'OpenCms..... | 25 |
| 3.3.1. Estructura d'OpenCms Workplace..... | 25 |
| 3.3.2. Creació de continguts..... | 32 |
| 3.3.3. Edició de continguts..... | 36 |

| | |
|-------------------------------------------------------|----|
| 3.4. Elements d'OpenCms | 38 |
| 3.4.1. Els mòduls | 38 |
| 3.4.2. Les plantilles o "Templates" | 41 |
| 3.4.3. Les etiquetes JSP | 42 |
| 3.4.4. Els Scriptlets JSP | 43 |
| 3.5. Creació d'un microsite..... | 43 |
| 3.5.1. Què és un microsite? | 43 |
| 3.5.2. Les plantilles d'un microsite..... | 44 |
| 3.5.3. Les pàgines d'un microsite | 45 |
| 3.5.4. L'ús de recursos a les plantilles | 45 |
| 3.5.5. L'ús d'elements externs a les plantilles | 46 |
| 3.5.6. Separació de parts de les plantilles | 46 |
| 3.5.7. Obtenció del lloc web | 48 |
| 3.6. Desar i recuperar dades..... | 48 |
| 3.6.1. Exportar de la base de dades..... | 49 |
| 3.6.2. Exportar un mòdul | 49 |
| 3.6.3. Importar un mòdul | 50 |
| 3.6.4. Importar a la base de dades..... | 50 |
| 3.7. Característiques especials d'OpenCms..... | 50 |
| 3.7.1. Continguts editables | 50 |
| 3.7.2. Tipus predefinits de dades | 51 |

Capítol 4

| | |
|---------------------------------------------------------------|----|
| Desenvolupament del projecte | 52 |
| 4.1. Anàlisi de requisits | 52 |
| 4.2. Disseny | 59 |
| 4.3. Prototipat | 66 |
| 4.4. Implementació | 70 |
| 4.4.1. Creació de l'espai per al lloc web dins d'OpenCms..... | 71 |
| 4.4.2. Creació dels primers elements bàsics | 72 |
| 4.4.3. Optimització del codi | 73 |

| | |
|---------------------------------------------------------------|-----|
| 4.4.4. Gestió d'idiomes..... | 75 |
| 4.4.5. Creació de les pàgines | 78 |
| 4.4.6. Utilització dels recursos..... | 81 |
| 4.4.7. Creació dels elements de navegació | 88 |
| 4.4.8. Introducció de noves funcionalitats | 92 |
| 4.4.9. Revisió del full d'estil..... | 97 |
| 4.5. Llançament | 97 |
| | |
| Capítol 5 | |
| Conclusions i futures extensions | 99 |
| | |
| Bibliografia..... | 100 |
| | |
| Annex A | |
| Guia d'administració dels continguts del lloc web GRIHO | 102 |

Capítol 1

Introducció

1.1. Introducció

Durant la realització dels meus estudis d'Enginyeria en Informàtica a la Universitat de Lleida (UdL) vaig decidir cursar l'assignatura optativa “Treball integrat en un grup d'Investigació” al Grup de Recerca en Interacció Persona-Ordinador i Integració de bases de dades (GRIHO) d'aquesta universitat.

GRIHO és un grup de recerca interdisciplinar i interdepartamental que té com a objectius principals la recerca, la docència i la transferència tecnològica a les empreses en la disciplina coneguda amb el nom d'Interacció Persona-Ordinador (IPO).

La IPO és la disciplina que estudia els intercanvis d'informació entre les persones i els ordinadors per a que totes les persones puguin realitzar les seves tasques d'una manera eficient i satisfactòria i amb un mínim esforç d'aprenentatge.

És en aquest marc en el qual vaig realitzar el meu Treball Final de Carrera (TFC) de l'Enginyeria Tècnica en Informàtica de Gestió titulat “Estudi d'Accessibilitat dels llocs web de la ciutat de Lleida”, l'any 2005.

El fet d'haver treballat amb aquests temes em va portar a decidir-me per col·laborar amb el grup de recerca per tal de millorar i consolidar els meus coneixements sobre la IPO.

En manifestar la meua intenció de col·laborar en el grup, de seguida va sorgir la possibilitat d'iniciar aquest projecte: redissenyar la web de GRIHO, treball mitjançant el qual podria aprofundir en el model de procés definit en la Tesi doctoral d'en Toni Granollers titulada “MPIu+a. Una metodología que integra la ingeniería del software, la integración persona-ordenador y la accesibilidad en el contexto de equipos de desarrollo multidisciplinares”, ahora que també podria conèixer el funcionament d'una nova tecnologia, el sistema gestor de continguts OpenCms.

1.2. Objectius

Els motius fonamentals que porten a la proposta de redissenyar la web del grup de recerca són els canvis que s'han produït a GRIHO, fet que ha convertit el lloc web en un espai que no en reflecteix les seves característiques, i difícil d'actualitzar i mantenir.

- En relació amb els membres que integren GRIHO, en el moment de crear la web antiga, el grup estava format per pocs professors i molts alumnes que principalment hi realitzaven els seus treballs de final de carrera.
Actualment, a més dels estudiants que hi continuen treballant, el nombre de professors que hi estan vinculats, molts dels quals són doctors, ha augmentat considerablement, i també hi col·laboren persones d'altres disciplines.
Així, en el nou lloc web s'ha d'aconseguir mostrar la realitat d'aquest nombrós grup d'integrants, permetent alhora facilitar la gestió de les seves dades.
- Pel que fa a la informació que ofereix, els objectius i els àmbits de treball del grup han canviat. Les línies de recerca en les quals es treballa són més clares i estan més definides.
Facilitar tota la gestió d'aquesta informació i continguts de les pàgines del lloc web és un dels objectius principals, que comporta la necessitat d'utilitzar un sistema gestor de continguts.
- La imatge del lloc web també és un punt que cal millorar, oferint un disseny que actualitzi el seu aspecte visual, mantenint-ne però alguns dels seus trets característics, com el logotip o els colors bàsics que identifiquen el grup (el blau i el taronja).
- I finalment, en relació a la plataforma tecnològica que allotja el lloc web, es tracta d'un servidor de GRIHO, un servidor que ha de mantenir el propi grup i que comporta que s'hi hagi de dedicar un nombre important de recursos, sobretot temps.

Aprofitant la possibilitat que ofereix la UdL, que disposa d'un servidor on es poden allotjar totes les webs de la universitat, es crearà la nova web de manera que es pugui encabir en aquesta plataforma.

En conclusió, l'objectiu és actualitzar el disseny de la web, definir nous continguts i reclassificar la informació que es vol mostrar, i alhora utilitzar una plataforma que permeti agilitzar la gestió d'aquesta informació.

1.3. Temporalització

Aquest TFC es va iniciar a finals del mes de gener del 2007, coincidint amb la meua entrada com a col·laboradora en el grup de recerca GRIHO, ja que en la definició dels objectius del meu treball amb el grup aquest era el punt fonamental.

Inicialment, vaig començar a treballar amb OpenCms, el gestor de continguts que seria la plataforma tecnològica per a la nova web, instal·lant tot el programari necessari i llegint tota la documentació per tal d'aprendre al màxim quin era el funcionament d'aquest sistema.

Paral·lelament, van iniciar-se els contactes amb els membres del grup per definir els objectius del treball i començar el model de procés per redissenyar el lloc web.

A mitjans de febrer, ja s'havia decidit la informació que hauria de proporcionar la web i quina estructura es seguiria, alhora que s'havia realitzat un esforç important amb OpenCms per aprofundir en totes les funcionalitats que oferia i que serien interessants per aplicar en el desenvolupament del treball.

Durant els mesos següents, es van crear els prototips necessaris que definirien la imatge de la web i les seves funcionalitats, tenint en compte la usabilitat i l'accessibilitat, i es va iniciar la implementació sobre la plataforma OpenCms.

Finalment, al mes de juny, el lloc web ja oferia totes les funcionalitats definides en els objectius i es va procedir al trasllat cap al servidor de la UdL, per tal d'iniciar el procés de llançament de la web i permetre que els membres de GRIHO poguessin començar a introduir-hi els continguts.

Així, el temps dedicat al desenvolupament d'aquest treball es va ajustar al calendari previst inicialment, fent coincidir la conclusió de la tasca amb el final del curs acadèmic 2006-2007.

1.4. Estructura del document

El document presenta un total de 5 capítols i un annex.

En el primer capítol es fa una breu introducció del treball, s'hi descriuen els seus objectius principals, i es presenta la temporalització del projecte.

En el segon capítol es descriu el mètode seguit per al desenvolupament del projecte, anomenat MPIu+a, es fa una introducció a l'eina que servirà per a la implementació, és a dir, un sistema gestor de continguts, i es fa un recull dels diferents coneixements que s'han hagut d'adquirir per poder fer el treball.

En el següent capítol s'aprofundeix en les característiques de la plataforma tecnològica utilitzada, el sistema gestor de continguts OpenCms, donant a conèixer les seves possibilitats i com s'utilitzen.

En el quart capítol s'explica tot el procés que s'ha seguit per desenvolupar el lloc web.

I en el darrer capítol es presenten les conclusions i futures extensions.

Finalment, s'adjunta un annex amb la informació que servirà de guia per a l'administració dels continguts del lloc web.

Capítol 2

Estat de l'art

2.1. Tecnologies relacionades

2.1.1. MPlu+a

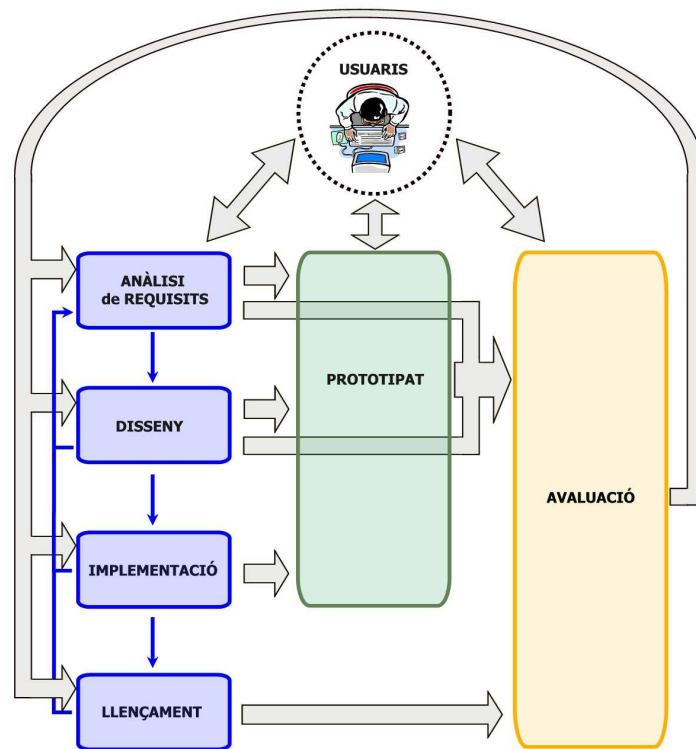
MPlu+a és el nom que rep el Model de Procés de l'Enginyeria de la usabilitat i l'accessibilitat.

Es tracta d'un mètode per al desenvolupament de sistemes interactius que intenta oferir una manera clara d'entendre tot el procés de desenvolupament i situar a l'usuari en cada moment dins d'una fase concreta.

Les activitats que determina aquest model són les següents:

- Activitats estructurades dels anàlisis dels requisits de la usabilitat.
- Un conjunt d'activitats explícit d'objectius d'usabilitat.
- Activitats de suport a una aproximació estructurada del disseny de la interfície d'usuari.
- Activitats d'avaluació dels objectius d'usabilitat mitjançant iteracions –cap a aquests objectius– en el disseny.

I l'esquema del model és el següent:



Aquest Model de Procés permet ajuntar el model lineal del desenvolupament de sistemes interactius de l'Enginyeria del Software amb els principis bàsics de l'Enginyeria de la Usabilitat incorporant una metodologia que guiï al desenvolupador en el procés de construcció d'un determinat sistema.

És per aquests motius que l'esquema incorpora els conceptes de prototipat i avaluació, i no té un sentit lineal ni restrictiu.

La funcionalitat de cadascun dels mòduls o etapes de l'esquema del MPIu+a es descriurà més detalladament en el capítol de Desenvolupament del projecte, a mesura que s'expliqui cadascuna de les fases del desenvolupament.

2.1.2. Sistemes gestors de continguts

Un Sistema Gestor de Continguts, en anglès “Content Management System” (CMS), és un entorn digital que permet emmagatzemar, gestionar i editar documents.

Pel que fa a l'emmagatzemament de documents, vol dir que el CMS escriu el document en algun tipus de memòria persistent, ja sigui en una base de dades, en un sistema de fitxers, o bé, en algun altre tipus de sistema.

Les funcions dels sistemes gestors de continguts són les següents:

- La majoria de sistemes gestors de continguts són la principal interfície per guardar i recuperar arxius que han emmagatzemat. Mentre que un processador de textos convencional escriu la informació en un sistema de fitxers que permet que els mateixos fitxers es puguin moure a diferents mecanismes d'emmagatzemament, de manera que es podran modificar o esborrar a través d'alguna altra aplicació, mentre que una aplicació CMS generalment requereix que tot l'emmagatzemament i la recuperació de documents es faci a través de mateix CMS.
- La segona funció d'un CMS és gestionar els documents. El CMS gestiona els continguts fent d'intermediari entre el sistema d'emmagatzemament i l'usuari. Els CMS disposen de motors de cerca per buscar documents, que poden manegar les metadades, com qui ha creat un document o quina ha estat la data de la última modificació, i la majoria permeten gestionar els documents de diferents maneres per a diferents usuaris.
- La tercera funció dels CMS és l'edició. Els CMS proporcionen als usuaris qualificats la possibilitat d'afegir, modificar i esborrar documents. Alguns CMS utilitzen eines externes per a l'edició, com processadors de textos, mentre que altres proporcionen les eines d'edició juntament amb l'aplicació CMS.

A més d'aquestes tres funcions generals, la majoria de CMS proporcionen eines de workflow, publicació i lliurament.

Workflow es refereix al procés estructurat o semi-estructurat de crear i editar un document, és a dir, quan més d'una persona treballa amb el document es pot automatitzar el procediment de moure el document a través de les seves fases de desenvolupament.

La publicació es refereix al procés de fer que un document estigui disponible per a un usuari, és a dir, el que bàsicament seria marcar-lo com a “fet” o “completat”.

El lliurament està relacionat amb la publicació i tracta el procés de moure un document al lloc desitjat, en aquest cas, publicar serà fer que el document estigui disponible per a poder ser vist a l'explorador web de l'usuari.

Els sistemes gestors de continguts es poden classificar segons l'entorn al qual van destinats tenint en compte diferents tipus d'objectius:

1. Si l'objectiu és orientar el CMS al mitjà

Habitualment, un sistema gestor de continguts es dissenya per tractar algun tipus particular de contingut que sol ser dictat pel mitjà cap al qual el CMS va enfocat. Per exemple, si el mitjà al qual va orientat el contingut és un publicació impresa, llavors els tipus de continguts que el CMS tractarà en general seran imatges i text amb format.

Per altra banda, si el mitjà al qual va orientat és la web, el CMS s'haurà d'organitzar per tal que pugui ajudar a preparar els continguts per a ser entregats online i tractarà principalment amb continguts que siguin interpretats pels clients, com HTML, formats d'imatge compatibles amb web (GIF, JPEG, PNG), CSS, i XML.

2. Si l'objectiu és orientar el CMS a la mida

Una segona manera de classificar els sistemes gestors de continguts és per la seva mida (el nombre d'usuaris, el nombre d'editors, el nombre de connexions simultànies, ...). Les mesures més habituals són el nombre de documents emmagatzemats i el nombre d'editors que treballen sobre el CMS.

Alguns sistemes CMS treballen amb “sites” (o llocs web) petits, amb poca gent que edita continguts i on el nombre de documents no passa del miler. Els sistemes que es troben dins d'aquest rang tendeixen a posar més èmfasi en la facilitat d'ús, més que en el suport de workflow, edició avançada o robustesa. Com que estan optimitzats per a llocs web petits no permeten escalar bé i el fet d'afegir múltiples editors pot comportar problemes de gestió i d'emmagatzemament que converteixen el sistema en inusable.

Els CMS enfocats a llocs web de mida mitjana permeten diversos editors amb diferents rols que, fins i tot, es poden organitzar amb grups o unitats. Estan construïts per manegar desenes o centenars de milers de documents, i intenten equilibrar la facilitat d'ús amb característiques que suportin cicles complexos de desenvolupament de continguts.

Algunes de les indispensables són el suport a workflow, versionatge, suport per a la publicació i diferenciació entre tipus d'usuaris.

Els CMS orientats als llocs web grans solen necessitar un desenvolupament fet a mida, degut al gran nombre de documents per gestionar i a la complicació de que pot suposar el procés editorial.

3. Si l'objectiu és orientar el CMS al model

La tercera manera de classificar els sistemes gestors de continguts és segons el seu model d'organització, que descriu com els usuaris es relacionen amb el CMS.

Els dos models més coneguts són el Model de Comunitat i el Model Proveïdor-Consumidor.

En el Model de Comunitat, el CMS està al centre i tots els continguts provenen de la comunitat, per tant, els membres tenen més facilitat per accedir al manteniment de continguts i disposen d'un espai personal per a contribuir als continguts.

En el Model Proveïdor-Consumidor, el CMS es troba entre un grup que proporciona el contingut, i un altre grup que l'utilitza. El primer grup té privilegis d'accés al CMS per a crear i mantenir els documents a través del CMS, i el segon grup, els usuaris o visitants, poden veure els continguts, i en alguns casos, generar alguns continguts com comentaris.

Per tal de complir amb els objectius exposats en la Introducció d'aquest treball, la plataforma tecnològica escollida és el sistema gestor de continguts OpenCms, ja que és el sistema que utilitza la UdL i que permetrà assolir l'objectiu final de crear una nova web per al grup de recerca GRIHO.

Dins dels tipus de sistemes gestors de continguts que s'han descrit, es pot situar OpenCms en el marc següent:

| | | mida | | | | |
|-------|---------------|--------|---------|------|----------------------|-------|
| | | petita | mitjana | gran | | |
| mitjà | imprès | | | | Comunitat | model |
| | digital (web) | | OpenCms | | Proveïdor-Consumidor | |

En el cas d'OpenCms, el mitjà al qual està orientat és la web. En realitat, OpenCms és un sistema gestor de continguts web dissenyat per a mostrar continguts web. Això vol dir que OpenCms per sí mateix és una aplicació web, per tant, els editors i els administradors hi accedeixen a través del navegador, alhora que està dissenyat per a emmagatzemar i servir continguts web com HTML, CSS, imatges i XML.

Es tracta d'un CMS orientat als llocs web de mida mitjana.

I està orientat bàsicament al Model Proveïdor-Consumidor. Els editors i els administradors es poden connectar a l'entorn de treball i tenir un control complet sobre els continguts, mentre que els usuaris només tindran accés a les pàgines publicades i alguna possibilitat limitada de contribuir als continguts.

El capítol següent d'aquest treball està dedicat a OpenCms, per tractar amb més profunditat aquest sistema i poder conèixer millor totes les seves característiques i funcionalitats.

2.2. Estudis realitzats per fer el treball

A més d'adquirir coneixements sobre les tecnologies relacionades que s'expliquen a l'apartat anterior, per a poder dur a terme aquest treball he hagut d'aprofundir i posar en pràctica diverses matèries cursades en els meus estudis d'Enginyeria en Informàtica.

Bàsicament són els temes relacionats amb assignatures de programació i amb l'assignatura d'Interacció Persona-Ordinador (IPO) de l'Enginyeria Tècnica en Informàtica de Gestió.

Pel que fa al Model de Procés que s'aplica en el desenvolupament del lloc web, havia estudiat tots els seus fonaments teòrics i els havia aplicat en alguns casos pràctics durant el desenvolupament de l'assignatura IPO, però no ha estat fins a la realització d'aquest projecte que he aprofundit realment en els mètodes per portar a terme cadascuna de les etapes del model per aconseguir una bona documentació i assolir l'objecte final del treball.

En referència als temes de programació, en concret, la programació web, he hagut d'aprofundir en dues vessants: per una part la programació d'HTML aplicant els fulls d'estil (CSS), i per altra banda, la programació de Java Server Pages (JSPs).

Capítol 3

OpenCms

3.1. Introducció

OpenCms és un sistema gestor de continguts web enfocat a organitzacions de mida mitjana i basat en el model proveïdor-consumidor.

Permet afegir, manegar, administrar i editar continguts a través d'una aplicació web.

Aquesta aplicació web s'anomena OpenCms Workplace i proporciona totes les eines necessàries per al manegament del contingut. S'hi pot navegar pels fitxers, fer cerques, posar els controls d'accés i manegar els usuaris.

Els continguts nous per a les aplicacions web es poden crear a través d'un editor gràfic que té aparença de processador de textos, i els documents i les imatges es poden pujar des de l'ordinador de l'usuari cap al sistema de fitxers d'OpenCms.

En resum, és un sistema dissenyat per permetre la majoria de tasques en un entorn web.

Una altra de les tasques d'OpenCms és aconseguir que els continguts estiguin disponibles per als visitants que cerquen informació. Aquests usuaris no accediran al OpenCms Workplace, sinó que accediran als documents que hagin estat publicats i marcats per tal que estiguin visibles.

Algunes de les seves característiques principals són:

- Interfície basada en el navegador web

Els usuaris d'OpenCms, sigui quina sigui la seva tasca, només han de disposar d'un navegador web. I per als nous usuaris és fàcil d'aprendre, ja que es comporta com una típica aplicació Windows.

- Eina d'edició WYSIWYG

Amb aquesta eina no cal que ni els creadors ni els editors hagin de treballar directament amb HTML, sinó que proporciona un interfície gràfica d'usuari

(GUI) per a compondre els continguts. Aquest editor funciona correctament en navegadors web basats en Mozilla i amb Internet Explorer a partir de la versió 5.

- Workflow basat en rols

OpenCms permet que el flux de treball es basi en rols per manejar millor tot el procés de creació de continguts.

- Model de permisos

Els mètodes utilitzats per permetre o restringir l'accés als usuaris són una combinació de llistes de control d'accés (ACLs), permisos i rols.

- Publicació sofisticada

Es poden publicar els continguts a nivell de fitxer o a nivell de projecte, i sempre es comproven els enllaços per evitar problemes de dependència i publicar junts tots els arxius necessaris.

Els continguts es poden exportar a altres sistemes per a que siguin subministrats o es pot fer a través del propi OpenCms.

- Framework basat en mòduls

OpenCms proporciona un mecanisme de mòduls per afegir funcionalitats al sistema. Hi ha mòduls de documentació com tutorials i referències, i altres mòduls per tractar continguts més sofisticats com calendaris, notícies o motors de cerca.

3.1.1. Breu història d'OpenCms

La història d'OpenCms comença l'any 1999 amb el seu predecessor, el MhtCms, un projecte similar que no era de codi obert.

Alexander Kandzior va començar a manipular sistemes de gestió de continguts (CMS) el 1994 i va desenvolupar la primera versió d'OpenCms a finals dels 90 mentre

treballava en diversos projectes personals. El 1998, les peces van començar a funcionar conjuntament en un sistema gestor d'un únic contingut.

El març del 2000, l'OpenCms Group, un conjunt de persones que treballaven en el sistema, van alliberar OpenCms sota una llicència open-source.

Sota els auspicis de l'OpenCms Group, diverses empreses i individuals, van començar a contribuir en la formació d'OpenCms.

Al 2002, quan OpenCms arribava a la versió 5.0, alguns dels fundadors van dirigir-se cap a altres projectes i l'OpenCms Group es va dissoldre.

Per tal de continuar amb el desenvolupament del seu software, Alexander Kandzior va crear una empresa nova, Alkacon Software, que va prendre les regnes del projecte OpenCms.

OpenCms és un programari de codi lliure, desenvolupat sota la Lesser GNU Public License (LGPL), que és una llicència certificada com a Open Source Initiative (OSI) creada per la Free Software Foundation.

La LGPL dicta que tot el codi font d'OpenCms ha de ser de lliure distribució.

Per altra banda, els components externs com mòduls o fitxers JSP poden utilitzar diferents llicències, així els desenvolupadors poden decidir les restriccions per al seu codi.

Actualment l'empresa Alkacon Software manté el projecte i també proporciona codi nou, a més d'oferir contractes de suport, cursos i servei de consultes.

El 30 de novembre del 2006 es va llançar la versió 6.2.3 d'OpenCms, i el mateix dia del llançament, Alkacon Software va informar dels plans que es seguien per a la versió 7 del projecte, anunciant el llançament de la versió estable d'aquesta pel juliol del 2007.

3.1.2. La comunitat OpenCms i alguns llocs de referència

Com la majoria de projectes de codi lliure, OpenCms té una comunitat molt activa de desenvolupadors i usuaris. La majoria són desenvolupadors de software que contribueixen en el codi directament. Altres creen mòduls per proporcionar serveis i característiques addicionals. I la majoria només utilitzen el producte. Tots aquests tres grups participen en fòrums de discussió i llistes de correu sobre OpenCms.

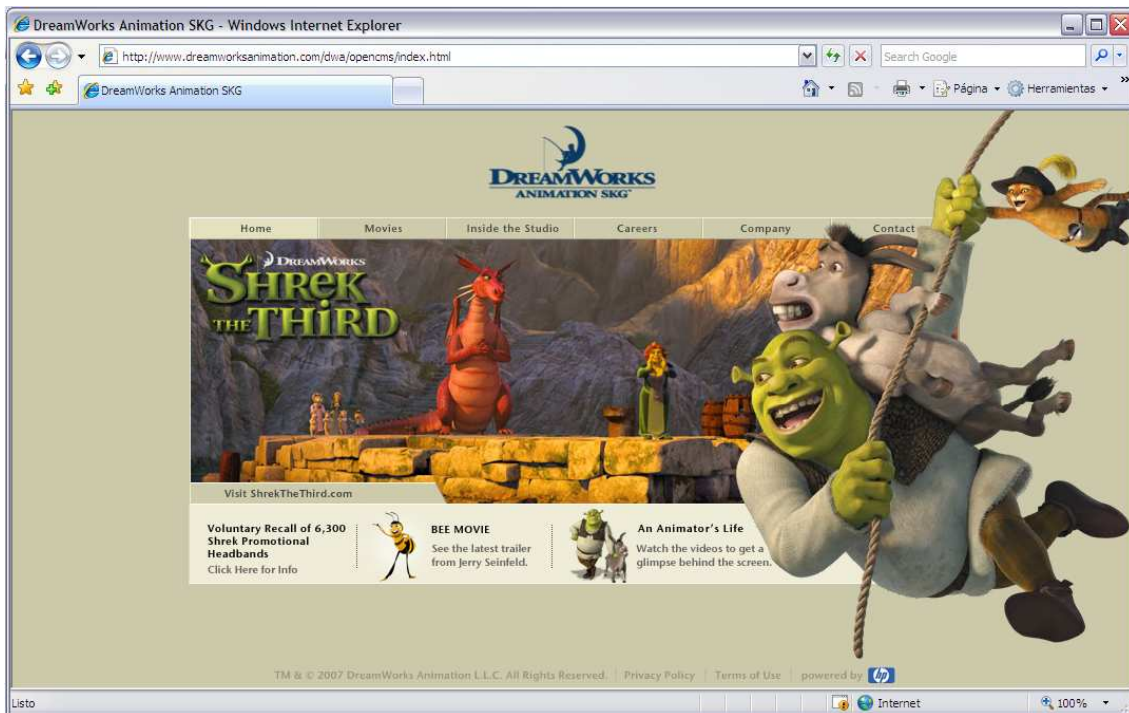
Amb molts contribuents, la llista de correu “opencms-dev” és un bon recurs per ajudar en la instal·lació, configuració i desenvolupament amb OpenCms, i una font molt útil de codi i nous mòduls.

Avui en dia, ja hi ha una gran quantitat de llocs web que funcionen amb OpenCms o l'utilitzen per al manegament dels seus continguts.

A més dels llocs web de nova creació de la UdL que l'utilitzen, també està present en nombroses universitats arreu del món, en organismes governamentals, bancs, empreses de telecomunicacions, ...

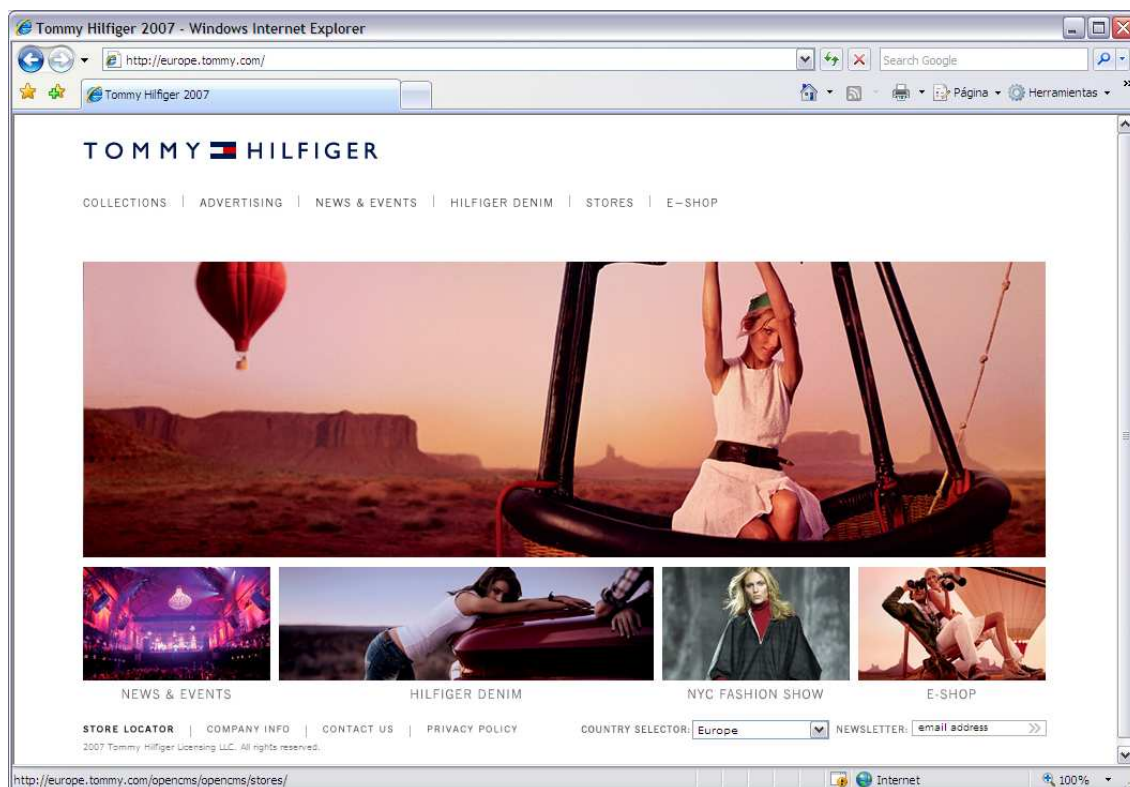
A continuació s'exposa una breu llista de llocs web de diversos àmbits per veure l'abast d'aquest sistema:

- Dreamworks Animation (<http://www.dreamworksanimation.com>)



- Banc Internacional d'Andorra i Banca Mora (<http://www.bibm.ad>)
- La Scala Opera House website (<http://www.teatroallascala.org>)
- The North Face (<http://www.thenorthface.com>)

- Tommy Hilfiger Global Sites (<http://europe.tommy.com>)



3.1.3. Descripció tècnica i components d'OpenCms

OpenCms està escrit en Java. Fa ús de l'estàndard XML i utilitza JDBC per emmagatzemar les dades en una base de dades relacional.

Com que està construït utilitzant Java pot funcionar en diferents plataformes, incloent nombroses versions de UNIX i Linux, i Windows.

Està dissenyat per a l'escalabilitat, per tant, funcionarà tant en un únic ordinador, com amb una col·lecció distribuïda de servidors.

Com que es tracta d'una aplicació basada en web, OpenCms funciona com un servlet Java dins d'un contenidor de servlets com Apache Tomcat o BEA WebLogic.

Per a l'emmagatzemament de les dades utilitza bases de dades SQL com MySQL, PostgreSQL i Oracle.

A continuació es mostra un breu resum de com funciona cadascun dels components:

- **El servidor web i els servlets Java**

El servidor web manega les connexions entrants. Passa les connexions d'OpenCms cap al contenidor de servlets per al processament. El contenidor de servlets manega un o més servlets Java. Mentre que un script CGI només funciona durant el temps necessari per processar una única petició, un servlet es manté funcionant fins que el servidor específicament l'atura (cosa que normalment només passa quan s'atura el servidor). La tasca del contenidor de servlets és proporcionar l'entorn run-time per als servlets.

Tot i que OpenCms es pot executar des de la línia de comandes, la majoria de vegades es fa com un servlet.

- **La base de dades**

OpenCms utilitza una base de dades per a l'emmagatzemament persistent de les dades. La informació sobre els tipus d'arxius, plantilles, i publicacions es guarda a la base de dades, i és tot el contingut.

OpenCms utilitza JDBC per connectar el servlet a la base de dades quan s'engega. El contingut es tracta dins la base de dades, a través de la qual es pot exportar cap a fitxers estàtics durant un cicle de publicació.

- **Pàgines, plantilles i pàgines Java Server (JSPs)**

El contingut s'emmagatzema en la base de dades en la forma de fitxers XML. La informació de la disposició i el codi de processament també estan a la base de dades, però no en els mateixos documents XML que els continguts.

Quan es fa la petició d'una pàgina, el contingut es treu de la base de dades i es posa en una plantilla. Qualsevol processament especial es fa en aquest punt, i llavors els resultats es tornen a qui fa la petició (que sol ser el navegador web) amb la forma d'un fitxer HTML complet.

Les plantilles i el codi personalitzat s'escriu en JSPs, un llenguatge estandarditzat per a encaixar instruccions de processament Java en un document HTML o XML.

3.2. Instal·lació d'OpenCms

OpenCms necessita una base de dades relacional configurada i un motor de servlets funcionant.

En aquest cas s'utilitza MySQL per a la base de dades i Tomcat per el motor de servlets.

Per al desenvolupament d'aquest projecte s'ha escollit la versió d'OpenCms 6.2.3, i a continuació es descriu el procediment seguit per portar a terme la instal·lació, indicant en cada punt el software utilitzat.

1. INSTAL·LACIÓ DE JAVA 2 SDK, A PARTIR DE LA VERSIÓ 1.4

OpenCms és una aplicació basada en web i s'executa com un servlet dins d'un contenidor de servlets, per tant, abans de poder executar cap aplicació Java serà necessària la instal·lació del Java System Development Kit.

És el primer pas necessari per a poder instal·lar OpenCms. En aquest cas s'ha optat per Java SDK en la versió 1.6.

2. INSTAL·LACIÓ DE TOMCAT

Tomcat és un motor de servlets desenvolupat conjuntament per Apache Software Foundation i Sun Microsystems. És de codi lliure i és la principal plataforma utilitzada pels desenvolupadors d'OpenCms.

OpenCms 6 necessita un contenidor que compleixi els estàndards Servlet 2.3 / JSP 1.2, i Tomcat 4 és la implementació de referència d'aquest estàndard.

S'ha instal·lat la versió 5.5 de Tomcat seguint el procediment indicat a la seva web.

3. INSTAL·LACIÓ DE MYSQL

MySQL és un servidor de base de dades relacional de codi lliure.

Tot i que OpenCms suporta altres bases de dades, tal com s'ha dit abans, els avantatges d'utilitzar aquesta són els següents:

- Funciona amb Windows i amb Linux
- És de codi lliure i gratuïta
- És la base de dades amb la qual treballen els desenvolupadors d'OpenCms

La versió recomanada per a OpenCms 6 és MySQL 5.0, per tant, és la que s'ha instal·lat.

4. DESPLEGAMENT DEL FITXER OPENCMS WAR

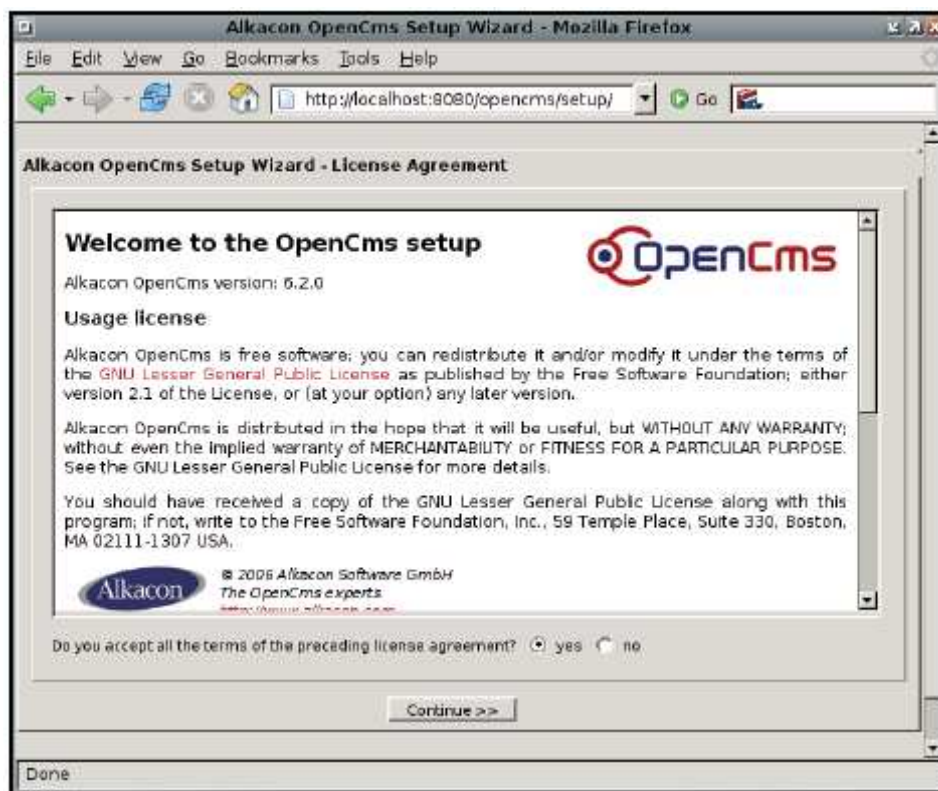
OpenCms, igual que la majoria de servlets Java, està empaquetat en un fitxer WAR (Web Archive). Un WAR és un tipus especial d'arxiu JAR que, al seu torn, és un tipus especial de fitxer ZIP, i conté tots els arxius necessaris per a l'execució d'una aplicació basada en servlets de Java.

Per a instal·lar `opencms.war`, s'ha de copiar aquest arxiu al directori `webapps` de Tomcat i seguidament s'ha de reiniciar Tomcat per a que aquest pugui desplegar l'aplicació OpenCms. Dins la carpeta `webapps` s'haurà creat el directori `opencms`.

5. INSTAL·LACIÓ D'OPENCMS

L'instal·lador està basat en web, així que cal obrir el navegador i anar a la URL <http://localhost:8080/opencms/setup/>, i apareixerà la pantalla d'instal·lació.

S'ha d'anar seguint les instruccions indicades a les pantalles, per tal d'indicar totes les dades que demani, que seran la bases de dades que s'utilitzarà, la qual inicialitzarà creant les taules corresponents, i s'importaran tots els mòduls disponibles com al sistema.



Un cop s'arribi a la pantalla de final d'instal·lació sense errors, ja es podrà accedir a <http://localhost:8080/opencvms/opencvms/system/login/> i entrarem al sistema OpenCms en la configuració per defecte amb l'usuari Admin i la contrasenya admin.

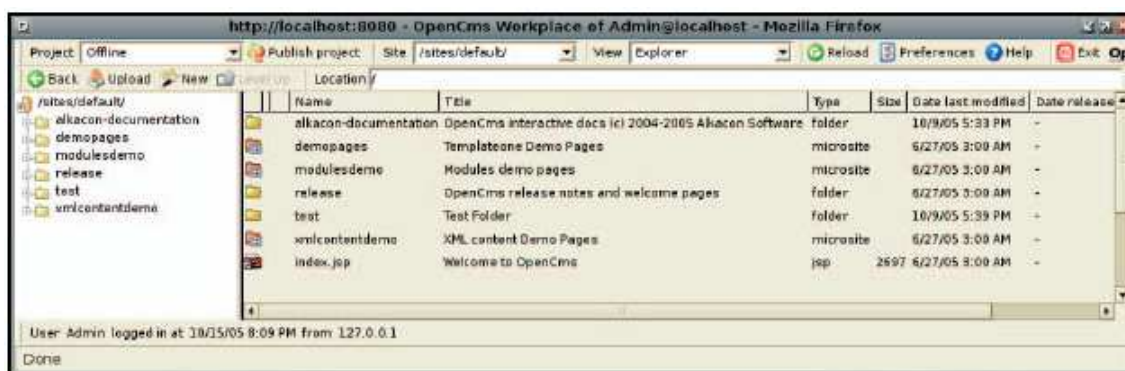


3.3. L'entorn de treball d'OpenCms

L'entorn de treball d'OpenCms s'anomena "OpenCms Workplace" i és la interfície primària per al manegament de continguts d'OpenCms. Té totes les eines per a l'edició de les pàgines, el manegament dels projectes, i l'administració del servidor.

3.3.1. Estructura d'OpenCms Workplace

Una vegada s'ha entrat a l'OpenCms Workplace, la primera pantalla que es mostra és la vista d'exploració (Explorer view).

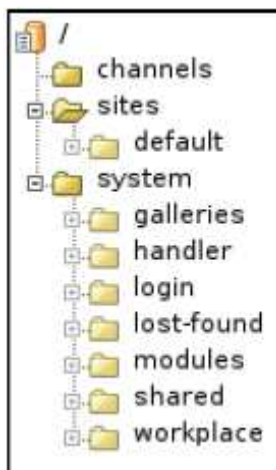


Amb aquesta vista es mostren els continguts del sistema de fitxers virtual o VFS ("Virtual File System"), com ens hi referirem d'ara en endavant.

Parts de la pantalla principal de l'entorn de treball:

- **El sistema de fitxers virtual o VFS ("Virtual File System")**

A la part esquerra de la vista d'exploració hi ha una pantalla en forma d'arbre de fitxers, que mostra totes les carpetes de fitxers de l'VFS.



L'VFS funciona igual que qualsevol sistema de fitxers, però emmagatzema les dades a la base de dades.

En alguns casos, però, els fitxers es poden copiar en el sistema de fitxers regular. Això es pot fer quan es fa la publicació, de manera que els fitxers seran còpies dels que es troben a l'VFS.

Si es clica sobre algun dels directoris d'aquest panell d'exploració de l'VFS, el seu contingut apareixerà al panell central de la finestra, el panell de detall de fitxers.

A la part superior de la jerarquia de l'VFS hi ha la carpeta arrel (/), que és la base.

La carpeta arrel té tres carpetes a dintre:

- **/channels**

S'hi emmagatzema la informació dels canals d'OpenCms. Els canals són una característica avançada per a organitzar els continguts.

- **/sites**

Conté un subdirectori apart per cada "site" que ha de servir OpenCms. Un "site", o lloc, és una unitat independent dins d'OpenCms, és a dir, no està relacionat amb altres sites i té el seu propi nom de domini i els seus propis conjunts d'editors o propietaris de continguts.

▪ **/sytem**

Conté elements que són importants per a la funcionalitat d'OpenCms. Té els següents directoris, cadascun amb unes funcions concretes:

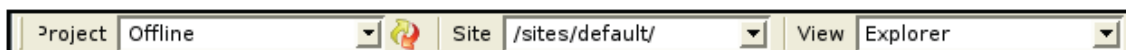
- ◆ **galleries:** Conté recursos compartits com imatges o fragments d'HTML que s'utilitzen amb freqüència.
- ◆ **handler:** Emmagatzema els arxius de suport d'algunes rutines que executa OpenCms en circumstàncies determinades, com per exemple quan s'han de mostrar pàgines d'error.
- ◆ **login:** Conté els fitxers per al management dels logins d'usuaris.
- ◆ **lost-found:** Si per algun motiu una pàgina es perd dins de l'VFS, anirà a parar a aquest directori..
- ◆ **modules:** És on es situen els mòduls nous d'OpenCms quan són instal·lats. Els mòduls serveixen per estendre la funcionalitat d'OpenCms.
- ◆ **shared:** Conté un parell de plantilles que són compartides pels recursos de diversos sites.
- ◆ **workplace:** És la darrera carpeta i conté tots els arxius (plantilles, imatges i continguts) que serveixen per construir el propi entorn de treball.

Tots aquests continguts del directori arrel s'utilitzen per a l'administració i el desenvolupament. La majoria d'usuaris que es dediquin a l'edició de continguts tindran l'accés restringit, de manera que només podran accedir als subdirectoris del directori `"/sites"`.

- **La barra d'eines**

A la part superior de la finestra hi ha la barra d'eines.

Començant per l'esquerra, hi ha tres llistes desplegable, la de Project, la de Site, i seguidament la de View.



La llista Project serveix per seleccionar en quin projecte es vol treballar. Per defecte està seleccionat el projecte “Offline”, que és el que conté tots els fitxers de l’VFS, mentre que els altres projectes seran, en certa manera, sub-projectes d’aquest.

Just al costat de la llista Project hi ha la icona corresponent al botó de publicació



a través del qual podrem publicar tots els recursos del projecte seleccionat.

La següent llista que apareix és la de Site, a partir de la qual podrem trobar tots els llocs web emmagatzemats en aquesta instància d’OpenCms. Per defecte hi apareixerà l’arrel de l’VFS “/”, i el lloc web per defecte “/sites/default”.

La darrera llista és la que permetrà que l’usuari seleccioni la vista amb la qual vol treballar. Hi ha tres possibilitats:

- **Explorer view**, que és la vista actual i permet fer la majoria de tasques d’edició.
- **Administration view**, que dóna accés a totes les eines d’administració tècnica i que sol utilitzar l’administrador dels sistema.
- **Workflow view**, la vista que proporciona a l’administrador dels projectes una interfície basada en tasques per al manegament dels projectes i dels continguts individuals.

Finalment, a la barra d’eines hi ha els botons:



- **Reload**, per recarregar la pàgina, tal com el seu nom indica, de la mateixa manera que ho faria el botó de refrescar del navegador habitual.
 - **Preferences**, per accedir a la pàgina de preferències, on l'usuari pot establir la seva configuració personal.
 - **Help**, que obre una nova finestra amb informació d'ajuda referent a la pantalla on estem.
 - **Exit**, que farà el log-out del sistema i ens portarà a la pantalla inicial per tornar a fer el log-in.
- **La barra de botons, també anomenada barra de navegació**

Es troba just a sota de la barra d'eines i conté botons relacionats específicament amb l'edició i l'exploració de continguts.



Igual que a l'explorador de Windows, hi ha el botó **Back**, per anar a la carpeta visitada anteriorment.

El botó **Upload**, que serveix per obrir la pantalla que permetrà carregar un fitxer del disc local al sistema de fitxers d'OpenCms, a l'VFS.

El botó **New**, que porta a la pantalla de creació de fitxers i permet crear qualsevol tipus de fitxer, incloses les carpetes.

El botó **Level Up**, que pujarà la localització al director pare del que ens trobem actualment.

Finalment, hi ha el quadre de text **Location**, que mostra la localització actual, relativa al site.

- **El panell de detall de fitxers**

En aquest panell es mostra el detall dels continguts de la ubicació de l'VFS seleccionada.

| | Name | Title | Type | Date last modified | Date released | Date expired |
|--|-----------------------|---------------------------------------------------------|-----------|--------------------|---------------|--------------|
| | alkacon-documentation | OpenCms interactive docs (c) 2004-2005 Alkacon Software | folder | 10/9/05 5:33 PM | - | - |
| | demopages | Template and Demo Pages | microsite | 6/27/05 3:00 AM | - | - |
| | modulesdemo | Modules demo pages | microsite | 6/27/05 3:00 AM | - | - |
| | release | OpenCms release notes and welcome pages | folder | 6/27/05 3:00 AM | - | - |
| | test | Test Folder | folder | 10/9/05 5:33 PM | - | - |
| | xmlcontentdemo | XML content Demo Pages | microsite | 6/27/05 3:00 AM | - | - |
| | index.jsp | Welcome to OpenCms | jsp | 6/27/05 3:00 AM | - | - |

Cadascun dels elements disposa d'un menú contextual amb les diverses accions que es poden realitzar sobre l'element, algunes de les quals poden no estar disponibles en diversos casos.



Les funcions que permeten cadascun dels elements del menú contextual són:

| | |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Lock / Unlock | Activa o desactiva, respectivament, el bloqueig d'un arxiu. És especialment útil quan hi ha diverses persones treballant amb un mateix projecte des de diferents ubicacions, per a evitar que diverses persones vulguin editar un mateix fitxer alhora. |
| Steal lock | Serveix per treure el bloqueig d'un altre usuari poder bloquejar el fitxer per a treballar-hi. |
| Publish directly | Publica només el fitxer seleccionat i els recursos dels quals |


| | |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | depèn, com poden ser imatges o plantilles. A més, si aquest fitxer té arxius germans (siblings), també es publicaran. |
| Edit page | Obre la pàgina en un editor. És l'eina principal per a crear continguts, i amb l'ús de l'editor s'evita haver d'escriure codi HTML directament, si així s'indica a l'editor. Aquesta funció només es pot utilitzar en els fitxers de tipus Page. |
| Edit | Serveix per editar els continguts d'alguns tipus concrets de recurs. Per exemple, "Edit controlcode" permet obrir una pàgina en l'editor de codi font per a poder editar l'HTML, l'XML o el JSP directament. Altres opcions poden ser "Edit" per a tipus especials, i "Edit link" per a editar recursos de tipus link. |
| Copy | Copia el fitxer. |
| Rename / Move | Canvia en nom d'un arxiu o el mou a un altre directori. |
| Delete | Esborra el fitxer. Cal destacar que aquest fitxer continuarà visible fins que sigui publicat de nou, moment en el qual s'esborrarà definitivament. |
| Touch | Marca un fitxer com a modificat, de manera que quan es torni a publicar el projecte, aquest es tornarà a copiar al projecte Online. |
| Undo changes | Torna a la versió anteriorment publicada del fitxer. |
| Show Siblings | Mostra tots els fitxers que estan declarats com a germans del fitxer seleccionat. |
| Permissions | Canvia els controls d'accés per al fitxer. Permet determinar quins grups o usuaris tenen permís per a llegir, escriure, publicar, controlar o veure el fitxer. |
| Secure / Export | Dóna accés a les configuracions de seguretat i exportació. |
| Change type | Permet canviar el tipus de fitxer, per exemple, de Text pla a JSP. S'ha d'anar amb compte, perquè alguns canvis de tipus poden comportar que alguns dels continguts no siguin tractats per OpenCms, per tant el seu comportament pot no ser el desitjat, tot i que no hi ha risc de pèrdua del contingut. |
| Change navigation | OpenCms pot construir la navegació del lloc web automàticament i amb aquesta opció, quan es crea un fitxer nou, |

| | |
|-------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| | es pot indicar quina posició ha d'ocupar en la navegació. |
| Edit controlcode | Proporciona accés directe al format XML amb el qual s'ha emmagatzemat el fitxer. És una opció útil en la resolució de problemes. |
| History | Mostra els canvis que s'han fet sobre un fitxer i permet accedir a versions anteriors del mateix. |
| Properties | Dóna accés a informació sobre el document com el títol, les paraules clau o la posició de navegació. |

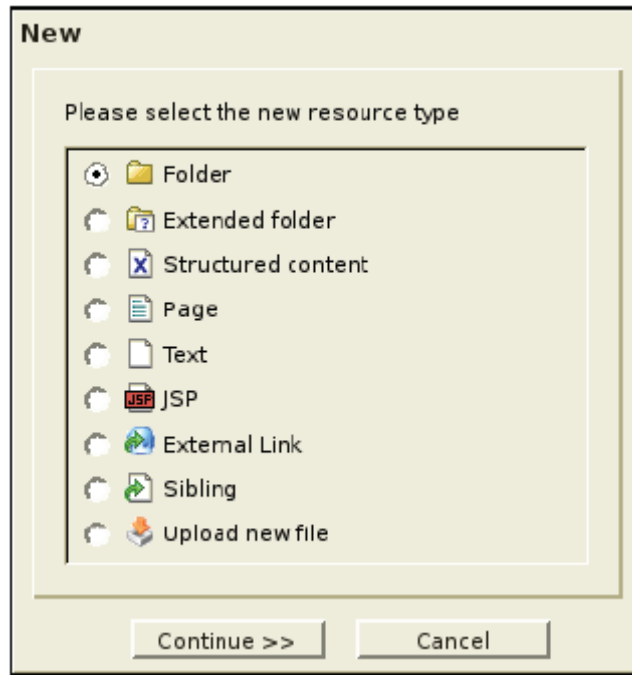
3.3.2. Creació de continguts

Després de veure com es navega a través de l'VFS, anem a veure com es creen els fitxers.

Per a crear un fitxer ens hem de situar, en el panell de detall de fitxers, dins de la carpeta on volem posar-lo i clicar sobre la icona que hem descrit a l'apartat "Estructura d'OpenCms Workplace" que serveix per crear nous continguts (la tercera icona de la

barra de navegació, el dibuix de la qual és una vareta màgica)  .

Apareixerà la pantalla de creació de fitxers on s'ha de seleccionar el tipus del nou recurs que volem crear.



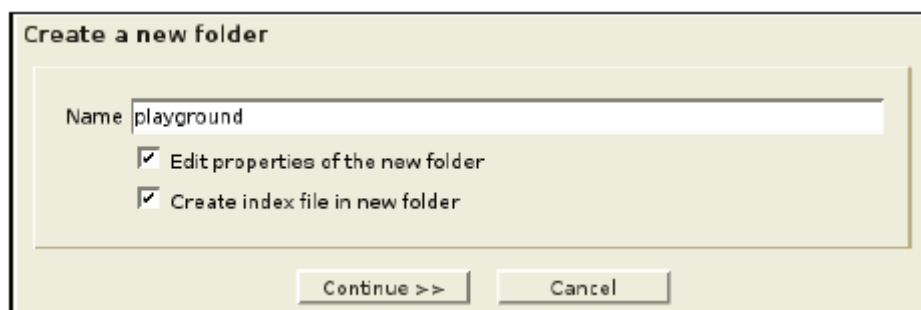
Els tipus de fitxers són els següents:

- **Folder**, per crear una nova carpeta al sistema de fitxers.
- **Extended Folder**, que és una variant del tipus Folder, ja que hi afegeix característiques addicionals segons el tipus d'Extended Folder que escollim.
- **Structured Content**, s'utilitza per crear documents amb caselles o continguts especialitzats que les pàgines normals no tindrien, com són l'hora, la ubicació o la informació de registre. Igual que en el cas anterior, si s'agafa aquesta opció s'haurà d'escollir el tipus de contingut estructurat.
- **Page**, és un document HTML que es pot editar amb l'editor WYSIWYG, que és l'editor de continguts per defecte d'OpenCms.
- **Text**, és el tipus genèric per a qualsevol contingut de només text.
- **JSP**, s'utilitza per a crear continguts dinàmics escrits com a JSPs.
- **External Link**, crea un punter des de la ubicació actual cap a un document extern d'OpenCms, posant-hi la URL completa.
- **Sibling**, crea una nova còpia germana d'algun fitxer existent, és a dir, permet establir una relació que faciliti el tractament de grups d'arxius de manera conjunta.
- **Upload new file**, proporciona una manera de seleccionar un fitxer de la nostra màquina i carregar-lo a dins d'OpenCms, creant un nou fitxer a l'VFS.

Segons el tipus de fitxer que es seleccioni apareixeran unes opcions o unes altres que caldrà completar. A continuació s'exposen breument dos exemples per veure quines poden ser aquestes opcions:

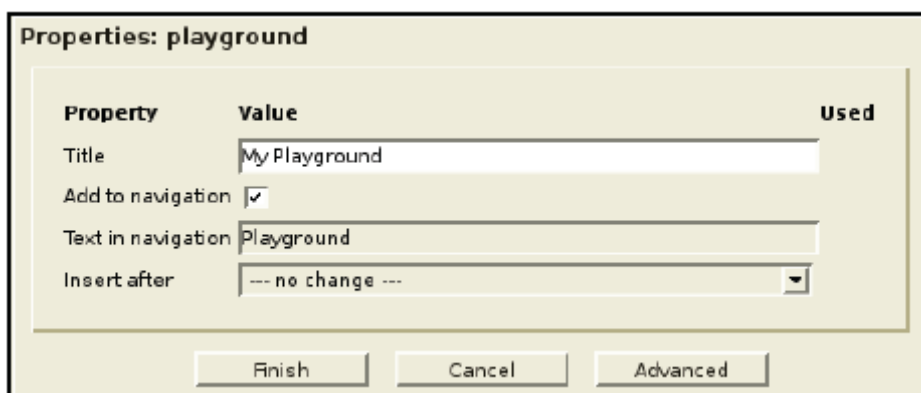
Exemple: creació d'una carpeta nova

Per exemple, si el tipus escollit és un Folder, és a dir, si volem crear una carpeta, apareixerà un diàleg on s'ha de posar el nom que li volem donar, i dos checkboxes per indicar si volem editar les propietats de la carpeta i si volem que també es creï un fitxer índex per a aquesta.



The image shows a dialog box titled "Create a new folder". It has a text input field for "Name" containing the word "playground". Below the input field are two checkboxes, both of which are checked: "Edit properties of the new folder" and "Create index file in new folder". At the bottom of the dialog are two buttons: "Continue >>" and "Cancel".

Si deixem els dos checkboxes seleccionats, tal com està establert per defecte i premem al botó per continuar, apareixerà la pantalla de propietats de la carpeta.



The image shows a dialog box titled "Properties: playground". It contains a table with three columns: "Property", "Value", and "Used".

| Property | Value | Used |
|--------------------|-------------------------------------|------|
| Title | My Playground | |
| Add to navigation | <input checked="" type="checkbox"/> | |
| Text in navigation | Playground | |
| Insert after | --- no change --- | |

At the bottom of the dialog are three buttons: "Finish", "Cancel", and "Advanced".

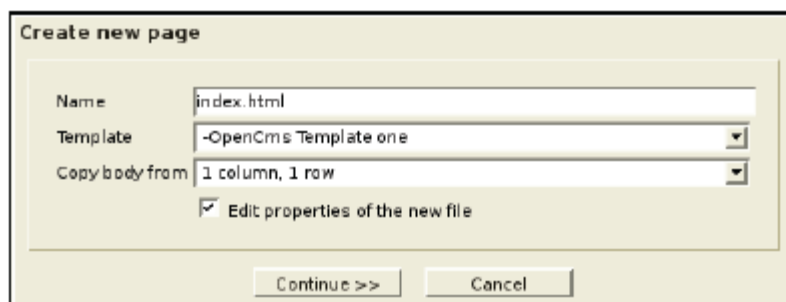
La primera de les propietats és el Títol. El títol de la carpeta és diferent del nom, ja que el nom s'utilitza com a part del path dins de l'VFS, el títol servirà per facilitar la interacció amb l'usuari.

Les propietats següents fan referència a la navegació del nou recurs creat dins d'OpenCms, és a dir, per indicar si volem que aparegui aquesta carpeta en els menús de navegació que es generen automàticament, i si és així, en quina posició i amb quina descripció hi han d'aparèixer.

Exemple: creació d'una pàgina

Seguint amb l'exemple anterior, en el qual s'ha indicat que es creï un fitxer índex per a la carpeta, quan s'acaben de completar les propietats d'aquesta, apareix el quadre de diàleg per a la creació d'una pàgina.

En aquest punt s'indica el nom del fitxer, que per defecte es diu "index" amb extensió ".html".



Llavors s'ha de seleccionar el "Template" per a aquesta pàgina, és a dir, la plantilla que utilitzarà, considerant que quan creem la pàgina només s'ha de crear el text bàsic, ja que la informació referent a la capçalera, el peu i la navegació estarà en un fitxer (plantilla) que estarà prèviament emmagatzemat, de manera que aplicant la plantilla a la pàgina aconseguirem un document completament formatat.

Durant el procés de creació d'una pàgina també es permet la configuració de les seves propietats, de manera similar a com succeeix quan es crea una carpeta.

3.3.3. Edició de continguts

A la part central de la finestra en la vista d'exploració, quan veiem el detall dels fitxers d'un directori, si obrim el menú contextual d'un dels fitxers, segons el seu tipus, podrem editar-lo amb algun dels tres tipus d'editors que proporciona OpenCms:

- **Els editors WYSIWYG**

Si es tracta d'un fitxer amb extensió HTML, per exemple, i escollim l'opció "Edit Page", s'obrirà la pàgina per a que pugui ser editada amb un editor WYSIWYG.

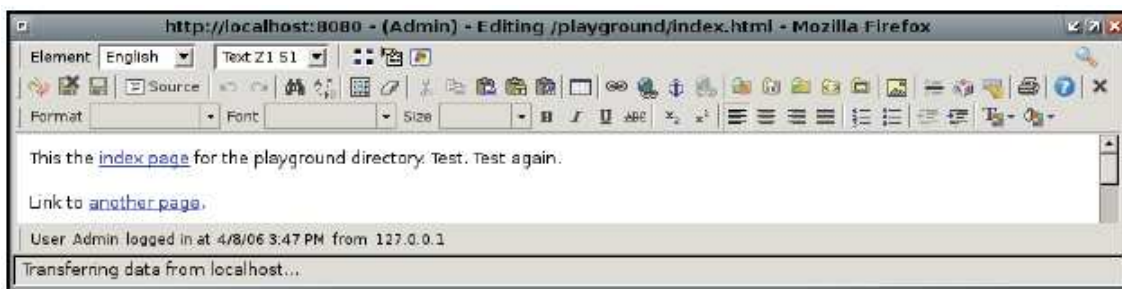
OpenCms 6 suporta tres editors WYSIWYG. L'editor per defecte és l'FCKEditor, mentre que els altres dos són l'editor Microsoft DHTML, que era l'editor per defecte en les versions anteriors d'OpenCms i que només funcionava amb Internet Explorer, i l'editor HTMLArea, un editor de text en línia de codi lliure escrit en Javascript, HTML i CSS, que ofereix les mateixes funcionalitats que l'FCKEditor i, per tant, en futures versions d'OpenCms desapareixerà, igual que l'editor Microsoft DHTML.

- **L'editor FCKEditor**

És el nou editor per defecte, que funciona amb Internet Explorer i també amb navegadors basats en Mozilla, com Netscape o Firefox.

És un editor que ha estat creat específicament per treballar amb OpenCms i és el recomanat pels desenvolupadors d'OpenCms.

La seva interfície és molt similar a la d'un processador de textos estàndard o a la d'una eina d'edició d'HTML.



Permet seleccionar en quin idioma es volen escriure els continguts; així es pot posar contingut per cada idioma disponible en una mateixa pàgina.


També permet dividir la pàgina en seccions de text que es podran editar seleccionant-les a través de la barra d'eines de l'editor, tenint en compte que s'identifiquen pel nombre de fila i columna on es troben de la forma "Text Z1 S2".

A la barra d'eines de l'FCKEditor hi ha molts més botons que faciliten l'edició del text alhora que permeten incloure altres elements com imatges, arxius adjunts, enllaços, ...

▪ Editar des d'una pàgina

A més de poder editar una pàgina accedint a l'editor a través del menú contextual del fitxer en la vista d'exploració, també s'hi pot accedir mentre es veu la pàgina que es vol editar en el navegador web.

Si hem entrat al sistema com a usuari amb permisos en l'entorn de treball, si a la pàgina que estem veient, en alguna de les seccions o elements hi apareix

la icona d'edició directa "Direct Edit button" , fent-hi clic es podrà editar aquesta part amb l'FCKEditor i desar els canvis.

• El Sourcecode Editor (o Editor de codi font)

El Sourcecode Editor és el que s'utilitza normalment per a fitxers de text i pàgines JSP, i també es pot utilitzar per escriure HTML directament.

En els fitxers HTML, si s'escull en el menú contextual l'opció per editar el codi font, s'obre l'editor FCKEditor, en el que clicant al botó "HTML Source" podrem accedir a l'edició directa del codi.

Per altra banda, aquest editor té una interfície molt senzilla però que tindrà totes les eines imprescindibles per a l'edició i desenvolupament de recursos de text, ja siguin continguts o scripts executables.

- **El Controlcode Editor (o Editor de codi de control)**

Aquest editor és un tipus especial de Sourcecode Editor.

Els continguts de les pàgines s'emmagatzemen en documents XML i en casos determinats pot ser necessari haver-los de modificar editant directament l'XML, enlloc d'utilitzar l'editor WYSIWYG.

3.4. Elements d'OpenCms

3.4.1. Els mòduls

En els apartats anteriors, per a la creació i edició de continguts, s'ha parlat de la vista d'exploració o "Explorer View" de l'entorn de treball d'OpenCms.

Tal com s'ha indicat, a OpenCms també s'hi pot treballar utilitzant la vista d'administració, seleccionant-la en el desplegable "Administration View".

Aquesta vista dóna les eines per administrar el servidor OpenCms, i una d'aquestes serveix per al manegament dels mòduls.

Com que OpenCms està dissenyat per a poder-se adaptar a l'entorn de diferents tipus d'organitzacions, els seus desenvolupadors van implementar una arquitectura de mòduls que es pot utilitzar per estendre OpenCms amb noves funcionalitats i continguts.

Els mòduls són col·leccions portables de codi i continguts.

Un mòdul d'OpenCms és un conjunt ben definit de carpetes i fitxers emmagatzemats en un arxiu ZIP. Un mòdul pot contenir fitxers JAR, classes de Java, scripts JSP, imatges, fulles d'estil, i qualsevol altre tipus de contingut que sigui suportat per OpenCms.

Els mòduls són una bon eina per empaquetar els recursos, en concret, les plantilles, els fitxers JSP i el codi Java.

- **Creació d'un mòdul**

En la pantalla de manegament de mòduls, si s'escull l'opció "New Module", s'inicia el procés de creació d'un mòdul nou.



En aquest punt s'han d'indicar, entre altres característiques, el nom del paquet i el nom del mòdul que volem crear, i s'han de seleccionar els directoris que es volen crear per defecte per a aquest mòdul, que es crearan automàticament a l'VFS d'OpenCms.

En la secció "Modules Folders" hi ha sis checkboxes per seleccionar els directoris.

Administration View > Module Management > New Module

New Module Up

Module information

Package name: ?

Module name: ?

Module description: ?

Module version: ?

Module group: ?

Action class: ?

Module creator

Author name: ?

Author email: ?

Module folders

Create Modulefolder: ? ☒

Create templates subfolder: ? ☒

Create elements subfolder: ? ☒

Create resources subfolder: ? ☒

Create classes subfolder: ? ☐

Create lib subfolder: ? ☐

Ok Cancel

Si s’escull l’opció “Create Modulefolder”, es crearà la carpeta per al mòdul al directori “/system/modules”, i serà en aquesta carpeta on estaran tots els fitxers del mòdul; per tant, sempre que s’hagi de crear un nou mòdul s’ha d’escollir aquesta opció.

Els altres cinc checkboxes seran subdirectoris d’aquesta carpeta principal, i les seves funcions són les següents:

- Subdirectori “templates”: per emmagatzemar les plantilles escrites en JSP.
- Subdirectori “elements”: per posar-hi parts de codi JSP o fitxers de text que siguin necessaris per al bon funcionament del mòdul.
- Subdirectori “resources”: per suportar arxius com imatges, JavaScript i fulles d’estil que utilitzaran les plantilles.
- Subdirectori “classes”: crear aquest subdirectori és una opció útil per als desenvolupadors que s’escriuen les seves pròpies classes de Java, ja que posant el aquest directori els fitxers .class compilats, aquests es carregaran quan s’engegui OpenCms.

- Subdirectori “lib”: per emmagatzemar fitxers JAR que continguin codi compilat amb recursos associats, és a dir, llibreries, que es carregaran en iniciar el servidor.

3.4.2. Les plantilles o “Templates”

Per entendre per a què serveixen les plantilles, anem a explicar què és el que passa quan un client demana una pàgina, és a dir, quan demana algun contingut.

El que succeeix és que OpenCms retorna els continguts de la pàgina demanada, que són bàsicament els continguts d'un fitxer en concret tal i com s'han creat a través de l'entorn de treball d'OpenCms.

Així doncs, aquests continguts proporcionats pel fitxer no porten informació associada que indiqui com s'han de distribuir dins de la pàgina del lloc web, ni la navegació corresponent a aquesta pàgina. Per tant, el fitxer haurà de tenir informació relativa a quina plantilla ha d'utilitzar, entre altres dades, per tal que es pugui obtenir l'efecte desitjat quan es retorna una petició de pàgina.

Les plantilles proporcionen l'estructura bàsica en la qual s'ha d'encabir el document, juntament amb la navegació i el format que s'ha de mostrar; de manera que quan els continguts s'adapten a la plantilla és quan el document es retorna al navegador del client.

Les plantilles es componen de dos tipus d'elements, codi HTML estàndard i continguts JSP. Els JSP serviran per a inserir continguts dinàmics dins dels fitxers HTML; i es pot fer de dues maneres: “tags” o “scriptlets”.

3.4.3. Les etiquetes JSP

Els tags o etiquetes de JSP tenen un aspecte similar al de les etiquetes HTML o XML. Quan un client demana un document JSP, el servidor llegeix el document per cercar tags JSP. Si en troba, executa el codi i insereix els resultats generats a la còpia del document que s'enviarà al client.

Els tags s'organitzen en llibreries de tags o “taglibs”. OpenCms proporciona una llibreria de tags que és molt útil per a treballar-hi, ja que permeten accedir i modificar continguts.

Alguns exemples de tags JSP en OpenCms són:

```
<%@ taglib prefix="cms" uri="http://www.opencms.org/taglib/cms" %>
```

L'etiqueta “taglib” situada al principi d'un fitxer indica al servidor que aquesta pàgina utilitzarà les etiquetes de la llibreria d'OpenCms. Les llibreries JSP s'anomenen per la seva URI, que en el cas de la llibreria de tags JSP per a OpenCms és “http://www.opencms.org/taglib/cms”.

```
<cms:user property="firstname"/>
```

En aquest segon exemple s'utilitza l'etiqueta “user” de la taglib d'OpenCms, que indica al servidor que agafi la informació d'usuari d'OpenCms i l'escrigui en la posició on apareix aquesta línia en el document.

Quan el servidor executa les etiquetes JSP, crea un document nou que només conté HTML que és el que s'envia al client.

3.4.4. Els Scriptlets JSP

Els scriptlets JSP són una altra manera de proporcionar continguts dinàmics en les pàgines JSP.

Els scriptlets són parts de codi Java que es poden inserir en un document; seria semblant a inserir codi JavaScript amb la diferència que en el cas dels scriptlets és el servidor, i no el client, qui executa el codi.

Un exemple d'scriptlet és de la forma:

```
<%  
org.opencms.jsp.CmsJspActionElement cms = new  
org.opencms.jsp.CmsJspActionElement(pageContext, request, response);  
out.println(cms.user("firstname"));  
%>
```

Els scriptlets s'escriuen entre els delimitadors `<% i %>`.

En aquest exemple es crea un nou objecte Java del tipus

“org.opencms.jsp.CmsJspActionElement”, anomenat “cms”, que té accés a totes les funcions de les tags JSP d'OpenCms.

El mètode “user” que apareix a la darrera línia d'aquest exemple realitza la mateixa funció d'agafar una informació d'usuari d'OpenCms que l'exemple de l'apartat anterior de Tags JSP.

La major part de l'API d'OpenCms es pot accedir a través dels scriptlets, per tant, es poden crear scriptlets que permeten fer manipulacions complexes de dades d'OpenCms.

3.5. Creació d'un microsite

3.5.1. Què és un microsite?

Una de les característiques d'OpenCms, pel que fa als “sites” o llocs web, és que permet manegar múltiples llocs web amb una única instància d'OpenCms.

En la configuració per defecte d'OpenCms, el directori arrel per al lloc web es situa al directori “/sites/default” de l'VFS.

Com que OpenCms té aquesta capacitat de manegament per a múltiples llocs, ofereix la possibilitat de crear un tipus especial de directori anomenat “Microsite” dins del directori arrel, que serà una unitat independent dins d’OpenCms, independent dels altres sites.

3.5.2. Les plantilles d’un microsite

En l’apartat anterior, “Elements d’OpenCms”, s’ha vist que els tags JSP, les llibreries de tags i els scriptlets proporcionen un mètode per barrejar continguts dinàmics en el codi de documents HTML, i la seva utilització principal en OpenCms és a les plantilles.

Amb els tags JSP es poden crear plantilles que generin informació de navegació, agafin continguts, i facin altres accions que el lloc web creat amb OpenCms sigui atractiu i entenedor per als visitants.

El primer pas per treballar amb plantilles és establir un espai per emmagatzemar els fitxers de plantilla; aquest espai són els mòduls.

El fet de posar les plantilles en un mòdul facilita la seva separació dels espais dedicats a guardar els continguts, i a més serà útil la possibilitat de poder-les moure d’un servidor a un altre conjuntament amb les seves imatges, fulles d’estil i fitxers de suport.

Per crear una plantilla nova, cal crear un fitxer JSP dins de la carpeta “templates” del mòdul que volem associar al microsite, i donar-li un nom i un títol, i un cop ja està creat el fitxer, s’ha d’accedir a l’opció d’edició del codi font (“Edit sourcecode”) a través del seu menú contextual.

La plantilla proporcionarà la disposició per als continguts de totes les pàgines que la tinguin associada.

El codi del fitxer plantilla podrà incloure tags JSP, scriptlets i, fins i tot, podrà importar altres fitxers JSP, tot i que en la seva majoria està compost per HTML.

3.5.3. Les pàgines d'un microsite

Cadascuna de les pàgines del sistema virtual de fitxers (VFS) d'OpenCms es guarda dins d'un tipus especial de fitxer XML que descriu el document.

Aquest XML pot incloure continguts en diferents idiomes i pot tenir diverses subseccions o “sub-documents” anomenats “elements” que podran ser accedits des de les plantilles.

Un exemple seria la utilització de l'etiqueta `<cms:include element="text1"/>` dins d'una plantilla.

L'element “text1” és l'element que es crea per defecte quan es crea un document nou, i amb l'etiqueta de l'exemple es mostraria el seu contingut en visualitzar la pàgina que tingui associada la plantilla que conté aquesta etiqueta.

Per crear una pàgina ens hem de situar al directori corresponent al nostre microsite i amb el botó “New” de la barra d'eines, indicar que volem crear un nou fitxer de tipus “Page”, al qual li donarem un nom i li associarem una plantilla, seleccionant en el desplegable “Template” la que volem que tingui aquesta pàgina.

Seguidament, es pot posar el contingut de la pàgina seguint les instruccions indicades a l'apartat “Creació i edició de continguts”.

D'aquesta manera s'aconsegueix que a les pàgines de lloc web només s'hi guardi la informació, és a dir, els propis continguts de la pàgina, independentment del format.

Finalment, es pot visualitzar la pàgina creada fent clic sobre el seu nom des de la vista d'exploració, i es mostrarà seguint el format indicat a la plantilla a la qual està associada.

3.5.4. L'ús de recursos a les plantilles

Els recursos s'utilitzen per tal de millorar l'aspecte de les pàgines. Tal com s'ha vist, les plantilles del microsite es guarden en un mòdul, i, com s'ha indicat anteriorment, els mòduls disposen d'un directori de recursos, anomenat “resources”, on s'hi poden emmagatzemar fulles d'estil, fitxers JavaScript i imatges.

Per tant, en el directori de recursos del mòdul on hi ha les plantilles del nostre lloc web s'hi pot crear un nou fitxer de text pla que contingui el full d'estil que utilitzaran les plantilles.

Per fer que una plantilla utilitzi un full d'estil, es fa ús de l'etiqueta `<cms:link>` d'OpenCms. Aquesta etiqueta permet agafar la ruta d'un fitxer de l'VFS i convertir-la en una URI que el navegador pot utilitzar per demanar el recurs en concret.

Per cridar el full d'estil des d'una plantilla s'ha d'incloure a l'etiqueta `<head>` o capçalera de la plantilla la instrucció:

```
<link type="text/css" rel="stylesheet"
href="<cms:link>/system/modules/[nom_del_modul]/resources/
[full_d_estil].css</cms:link>"/>
```

Així, sempre que es faci referència a un fitxer de l'VFS des d'una plantilla, per exemple quan es vulguin utilitzar recursos, serà imprescindible l'ús d'aquest tag per obtenir el resultat correcte.

3.5.5. L'ús d'elements externs a les plantilles

En algunes ocasions pot ser interessant guardar parts d'una plantilla en fitxers separats, per exemple, si podem reutilitzar part del codi per ser utilitzat en múltiples plantilles. Seria el cas del codi de navegació.

La millor manera de gestionar els elements externs és crear un fitxer JSP per a cadascun d'aquests scriptlets o porcions de codi, i emmagatzemar-los al directori "elements" del mòdul. Quan s'hagi d'incloure a la plantilla algun dels elements, es farà ús de l'etiqueta "include" d'OpenCms, afegint una instrucció del tipus

```
<cms:include file="../elements/navigation.jsp"/>
```

al lloc pertinent de la plantilla.

3.5.6. Separació de parts de les plantilles

Quan es vol crear un fitxer dins del microsite, de la mateixa manera que es pot crear una nova pàgina, també s'hi pot crear un document JSP que pot ser útil per manegar formularis o mostrar informació dinàmica.

Tot i això, la utilització de les plantilles per part de les pàgines JSP és diferent de la que fan els fitxers de tipus pàgina, és a dir, els fitxers amb extensió HTML.

Per tant, és necessari modificar l'estructura de les plantilles i dividir el seu codi en diferents parts.

- **Creació de seccions dins d'una plantilla**

Per separar el codi d'una plantilla ja creada en diverses seccions s'ha d'utilitzar l'etiqueta d'OpenCms `<cms:template>`.

Aquesta etiqueta és de la forma

```
<cms:template element="nom_element">
```

seguida del codi de la plantilla que ha de contenir aquesta secció que s'identifica amb el nom de l'element, i correctament tancada amb `</cms:template>`.

Per separar bé una plantilla en seccions caldrà que la totalitat del seu codi pertanyi a algun element.

Si la plantilla modificada està associada a un fitxer de tipus pàgina, els elements s'aniran processant per ordre d'aparició, per tant, els canvis no suposaran cap modificació si s'ha mantingut l'ordre original del codi.

- **Utilització de plantilles en els documents JSP**

Per utilitzar els elements d'una plantilla des d'un document JSP s'ha d'indicar dins de les propietats avançades del fitxer, a l'apartat anomenat "template", la ruta cap al fitxer plantilla situat al directori "templates" del mòdul corresponent dins de l'VFS.

El codi de la pàgina començarà amb les declaracions típiques de JSP per a importar la llibreria d'etiquetes cms:

```
<%@ page session="false" %>  
<%@ taglib prefix="cms" uri="http://www.opencms.org/taglib/cms"%>
```

I seguidament es podrà accedir als elements de la plantilla que s'ha dividit i que està associada a la pàgina amb instruccions del tipus:

```
<cms:include property="template" element="[nom_element]"/>
```

Quan es processa la pàgina, les etiquetes "include" es substitueixen pels continguts dels elements de la plantilla. Així, l'avantatge que això ofereix és que

no cal incloure tots els elements de la plantilla i es pot afegir codi nou entre els elements en format HTML, que crearan una pàgina resultant diferent.

3.5.7. Obtenció del lloc web

Amb la combinació de plantilles i recursos i l'associació d'aquestes plantilles sobre documents que pertanyen a un microsite es pot aconseguir crear un nou lloc web.

Elements externs com el codi de navegació i recursos com un bon full d'estil permetran crear de manera estructurada i senzilla la web.

3.6. Desar i recuperar dades

Tota la informació dels fitxers amb els quals es treballa amb OpenCms és emmagatzemada a la base de dades, per tant, en la base de dades sobre la que es treballa hi ha tota la informació de l'VFS, encara que es vegi el sistema de fitxers organitzat igual que qualsevol jerarquia de fitxers d'un sistema operatiu normal.

En la vista d'administració, dins de l'apartat de manegament de bases de dades, hi ha les eines necessàries per posar i treure dades de la base de dades.



Són unes eines molt útils, ja que faciliten la possibilitat de fer backups o còpies de seguretat, restaurar una base de dades antiga i també carregar continguts d'altres servidors d'OpenCms.

3.6.1. Exportar de la base de dades

Els recursos que estan emmagatzemats es poden seleccionar de manera selectiva per a ser exportats.

Exportar significa empaquetar els arxius en un arxiu únic que es copia al sistema real, ja que mentre es treballa amb OpenCms s'utilitza el sistema d'arxius virtual.

Es pot exportar tot el contingut de l'VFS o només una part en concret.

Les dades exportades van a parar a un fitxer comprimit ZIP que queda guardat en el sistema de fitxers del servidor.

Amb aquesta eina es pot fer un backup complet de tots els continguts de l'VFS incloent tota la informació del directori arrel (/) del sistema. Això s'aconsegueix marcant l'opció "Include System Folder".


Aquesta opció també s'ha de considerar si només volem exportar les dades d'un lloc en concret.

Tal com s'indicava a l'apartat que descriu l'estructura de fitxers d'OpenCms, els continguts es guarden als directoris que es troben a la carpeta /sites, però dades de gran importància que poden estar relacionades amb aquests continguts, com les emmagatzemades en els mòduls, es troben en una altra part de l'VFS, a /system/modules i per tant s'haurà de fer l'exportació des d'un altre punt del sistema d'administració, l'apartat de manegament de mòduls.



3.6.2. Exportar un mòdul

Dins de la secció dedicada al manegament de mòduls, apareix una llista amb tots els mòduls integrats o creats amb l'OpenCms.

Prement en la icona d'exportació d'un mòdul  es podrà guardar en un fitxer comprimit.

3.6.3. Importar un mòdul

Un mòdul que prèviament ha estat exportat, queda guardat en forma de fitxer comprimit a la ruta de tomcat “\$CATALINA_HOME/webbaps/opencms/WEB-INF/export/modules”. Quan s’importa un mòdul es sobreescrueixen tots els fitxers que hagin estat modificats i s’afegeixen els que s’hagin creat nous, esborrant aquells que no estiguin en la versió que s’importa.

Per a importar un mòdul s’ha de seleccionar la opció “Import Module with HTTP” i escollir el fitxer corresponent a aquest mòdul.

3.6.4. Importar a la base de dades

La importació de dades és semblant a la de mòduls, amb la diferència que els fitxers que no es trobin a la nova versió no seran eliminats.


Els fitxers de dades que hagin estat exportades es poden trobar a “\$CATALINA_HOME/webbaps/opencms/WEB-INF/export”, i per a importar-les s’ha d’accedir a l’apartat de manegament de bases de dades, escollir la opció “Import File with HTTP” i indicar l’arxiu que conté les dades.

3.7. *Característiques especials d’OpenCms*

3.7.1. Continguts editables

OpenCms disposa d’una opció anomenada “Direct Edit” que permet als usuaris autoritzats modificar els continguts d’una pàgina mentre l’estan visualitzant en el navegador.



A aquests usuaris els apareix el botó  que permet editar a les zones de la pàgina que estiguin habilitades a l'efecte.

És una característica d'OpenCms que facilita el manteniment dels continguts de la web, ja que permet editar sense la necessitat d'haver d'accedir a l'entorn de treball d'OpenCms.

Per habilitar aquesta opció cal que la plantilla que té associada la pàgina estigui ben configurada. Només cal que s'hagi afegit l'etiqueta `<cms:editable/>` de la llibreria de tags d'OpenCms a la capçalera, és a dir, dins del `<head>`.

Llavors, quan es vulgui introduir un element editable, s'ha de posar

```
<cms:include element="[nom_element]" editable="true"/>
```

en el lloc corresponent de la pàgina.

Així, quan l'usuari autoritzat premi el botó d'edició directa s'obrirà l'editor i podrà actualitzar els continguts.

3.7.2. Tipus predefinits de dades

Una altra característica interessant d'OpenCms és la possibilitat d'utilitzar fitxers de contingut estructurat.

Des de l'estàndard W3C XML Schema, OpenCms pot configurar un editor per a la introducció de continguts basat en formularis.

Els continguts estructurats són un tipus de fitxer d'OpenCms que serveixen per a crear documents amb camps o continguts específics.

La versió d'OpenCms utilitzada en aquest treball incorpora diferents tipus predefinits en els mòduls de demostració, com són els tipus Notícies o Esdeveniments, i que poden mostrar-se en diversos formats de sortida, per exemple, en forma de llista, o bé, en forma de pàgina detallada per cadascun dels elements; seran opcions facilitades per les instruccions de les etiquetes corresponents de l'OpenCms taglib.

Capítol 4

Desenvolupament del projecte

Per al desenvolupament d'aquest projecte s'han seguit els principis del Disseny Centrat en l'Usuari, en concret, les fases del Model de Procés de l'Enginyeria de la usabilitat i l'accessibilitat (MPIu+a).

L'inici del projecte va començar amb l'Anàlisi de Requisits del sistema, seguint amb la fase de Disseny, a través de la qual es va determinar, entre altres, l'arquitectura de la informació del lloc web i que va permetre iniciar l'etapa de Prototipat, amb diverses propostes tant de paper com de software, per poder acabar implementant el nou lloc web del grup de recerca Griho.

Cadascuna d'aquestes etapes ha estat acompanyada de diverses Avaluacions per aconseguir que el sistema resultant compleixi amb els requisits establerts i s'adapti a les necessitats del grup.

A continuació es detallen les etapes esmentades, explicant-ne breument la seva base teòrica, i els mètodes emprats per portar-les a terme.

4.1. Anàlisi de requisits

Analitzar els requisits d'un sistema interactiu suposa el determinar, enumerar i classificar totes les característiques, capacitats i restriccions que aquest ha de complir o als quals es veu sotmès.

Els requisits d'un sistema interactiu fins fa poc temps només feien referència al component funcional, mentre que ara es tenen més en compte els aspectes relacionats amb els usuaris i amb l'ús que fan dels sistemes.

En aquesta fase del model de procés s'intenten establir els requisits del sistema en les etapes inicials del seu cicle de vida, fent més èmfasi a la vista externa i a la vista de l'usuari, en lloc de referenciar només la qualitat interna o funcional.

L'objectiu de l'Anàlisi de Requisits és aconseguir la qualitat necessària i suficient per trobar les necessitats reals dels usuaris. Es tracta d'una tasca sovint complicada, ja que les necessitats que solen exposar els usuaris normalment divergeixen de les seves necessitats reals, alhora que aquestes necessitats van evolucionant amb l'ús del sistema.

Amb aquesta anàlisi es pretenen establir els serveis que el sistema ha de proporcionar i les restriccions sota les quals ha d'operar, és a dir, què ha de fer i com ho ha de fer.

Pel que fa a la recollida dels requisits de la nova web de Griho, s'han realitzat les següents activitats:

- **Anàlisi dels implicats (Stakeholders)**

Per definició, un implicat en una organització és qualsevol grup o individu que pot afectar o pot ser afectat per la consecució dels objectius de la organització.

En el cas de l'objecte d'aquest treball, els implicats seran totes les persones responsables del disseny i el desenvolupament del sistema, de la seva implantació i manteniment, i les persones que tinguin interès en el seu ús, és a dir, els usuaris. És per aquest motiu que els implicats en aquest cas són els membres del grup de recerca GRIHO.

Amb aquests implicats es fan diverses reunions per a recollir la informació sobre els motius de desenvolupament de la nova web i els objectius que s'han d'assolir.

- **Establiment dels objectius**

Bàsicament l'objectiu és actualitzar el disseny de la web, definir nous continguts i re-classificar la informació que es vol mostrar, i alhora proposar una plataforma que permeti agilitzar la gestió d'aquesta informació.

Els objectius extrets de les reunions amb els implicats (StakeHolders Meetings) són els que l'exposen a l'apartat d'objectius del capítol introductori d'aquest treball.

En recollir els objectius d'un lloc web s'han de tenir en compte tant els objectius funcionals, com els objectius d'usabilitat i accessibilitat.

- **Objectius funcionals**

Els objectius funcionals que s'han definit per aquest lloc web són:

- Redefinir la informació que es mostra per reflectir de manera més fidel l'activitat del grup de recerca.
- Actualitzar la informació referent als membres integrants del grup.
- Utilitzar un sistema que permeti agilitzar el manteniment i actualització dels continguts.

- **Objectius d'usabilitat**

La usabilitat és vista generalment per assegurar que els productes interactius siguin fàcils d'aprendre, efectius i agradables per als seus usuaris.

Per aconseguir-ho, s'intentaran optimitzar les interaccions que les persones duguin a terme amb el nostre sistema a través dels objectius d'usabilitat següents:

- Facilitat d'aprenentatge: la interfície ha de ser simple, amb funcionalitats accessibles i ben definides que siguin fàcils d'aprendre i d'utilitzar.
- Consistència: en el text s'evitaran els fons foscos i els colors massa vistosos, i en els vincles, s'utilitzaran colors diferents per distingir els que s'hagin visitat.
- Robustesa: s'evitarà l'ús de marcs (frames), ja que certs navegadors no els suporten, i normalment també resten accessible quan es treballa amb lectors de pantalla.
- Disminució de la càrrega cognitiva: l'usuari hauria d'assolir qualsevol pàgina en el menor nombre possible de clics del ratolí.

▪ Objectius d'accessibilitat

Que el lloc web validi els estàndards:

- d'accessibilitat definits per l'W3C¹, que actualment són majoritàriament acceptats per la comunitat internacional

*El compromís de l'W3C de guiar la Web al seu màxim potencial inclou arribar a un alt grau de funcionalitat per a persones amb discapacitats. La **Iniciativa d'Accessibilitat Web (WAI)**, creada per l'W3C, ha desenvolupat treballs dels quals se'n desprenen les “Directrius d'Accessibilitat”.*

¹ El Consorci World Wide Web (W3C) és un consorci internacional en el qual les organitzacions membres treballen conjuntament per a desenvolupar estàndards Web. La seva missió és guiar la Web fins al seu màxim potencial a partir del desenvolupament de protocols i pautes que assegurin el creixement futur de la Web.

*Aquestes directrius són considerades en la Unió Europea com a **normes “de facto”**, i són citades com a referència obligada en la majoria de les legislacions sobre Tecnologies de la Informació de tot el món.*

*Aquestes Directrius d'Accessibilitat del Contingut Web s'anomenen **WCAG**.*

- de codi HTML definits també per l'W3C

W3C ofereix eines per comprovar documents HTML en conformitat amb les seves recomanacions sobre HTML i altres estàndards HTML.

- de fulls d'estil CSS

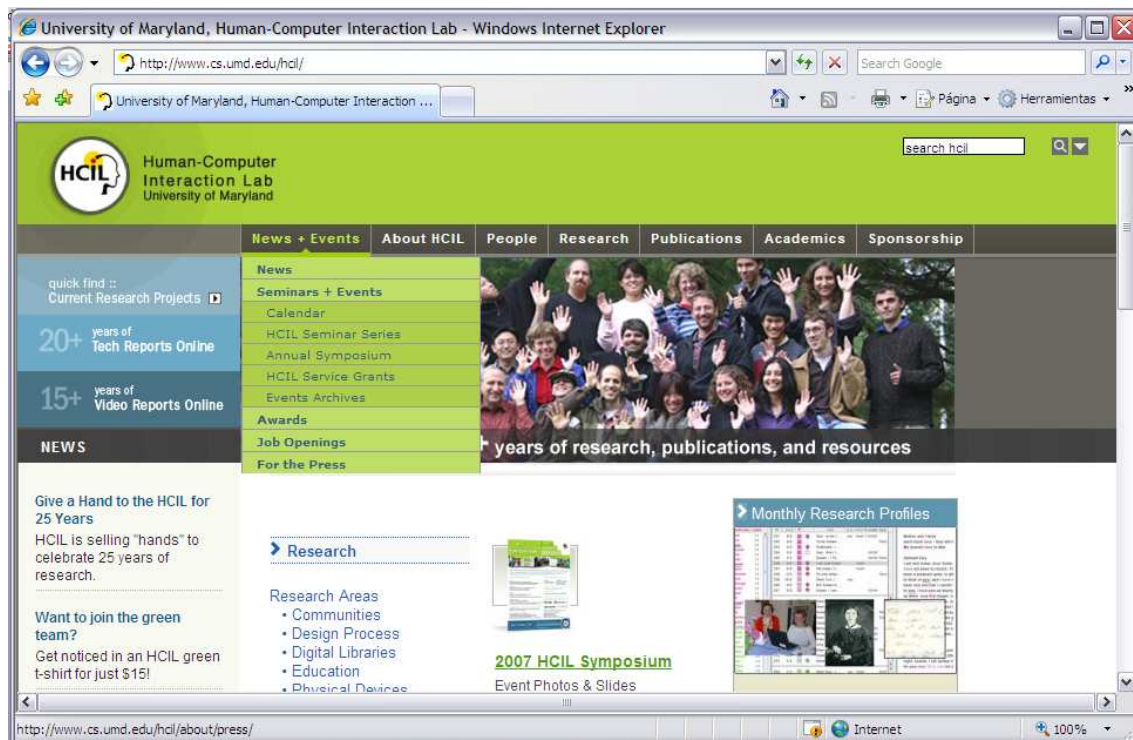
W3C també proporciona eines per validar la sintaxi dels fulls d'estil (CSS) que s'incrusten als documents HTML, recomanant sempre validar primer els HTML.

- **Anàlisi de la competència**

Per definir la informació que es voldrà transmetre amb la nova web, s'han analitzat i revisat tots els continguts que oferia la web antiga de Griho.

També s'han visitat els llocs web d'altres grups de recerca del mateix àmbit de Griho, com els següents:

- HCIL (Human-Computer Interaction Lab), de la Universitat de Maryland
(<http://www.cs.umd.edu/hcil/>)



- LoUISE (Laboratorio de Interacción con el Usuario e Ingeniería del Software),
de la Universitat de Castilla-La Mancha
(http://www.i3a.uclm.es/consulta/memoria.php?lang=es¶m_0=4¶m_1=62)



- GEDES (Grupo de Investigación en Especificación, Desarrollo y Evolución del Software), de la Universitat de Granada

(<http://lsi.ugr.es/~gedes/index.htm>)



- **Estudi de la nova plataforma**

La plataforma tecnològica proposada pels membres del grup per a allotjar-hi el sistema és el gestor de continguts OpenCms, ja que és la que s'utilitza per a la majoria de les webs de nova creació de la Universitat de Lleida. Per tant, s'han estudiat totes les possibilitats que ofereix aquest sistema i que poden ser aplicades al lloc web de Griho.

La possibilitat més avantatjosa d'OpenCms en funció dels objectius dels implicats és la simplificació de les tasques de manteniment de continguts.

Totes les característiques que ofereix OpenCms es poden trobar ben detallades al capítol anterior.

4.2. Disseny

En la segona fase del Model de Procés de l'Enginyeria de la usabilitat i l'accessibilitat (MPIu+a), dedicada al disseny del sistema, s'estableix l'arquitectura de la informació del lloc web i tota la part que fa referència al seu disseny gràfic.

En aquesta primera fase de disseny s'ha decidit l'arquitectura de la informació, mentre que la part de disseny gràfic, és a dir, l'aspecte visual de la web, no s'ha començat a tractar fins que no s'han obtingut els prototips que definirien l'estructura de les pàgines i s'explica, per tant, en l'apartat de Prototipat.

A partir de diverses reunions amb els implicats, els anomenats "Focus Groups", s'han recollit propostes pels continguts de la web i s'han establert quins hi haurà definitivament. Així, la proposta de continguts recollits en els Focus Groups és la següent:

| <u>Estructura proposada per al lloc web GRIHO en data 08/02/07:</u> | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> CATEGORIES DE 1er NIVELL: <p>En el Menú Principal:</p> <p>PRESENTACIÓ MEMBRES LÍNIES DE RECERCA PROJECTES INFRAESTRUCTURES PUBLICACIONS DOCENCIA LINKS RELLEVANTS</p> <p>Fora del Menú Principal:</p> <p>NOTÍCIES ESDEVENIMENTS (SEMINARIS, CONGRESSOS, ALTRES...)</p> <p><u>Propostes per a la pàgina d'inici:</u></p> <ul style="list-style-type: none"> Breu descripció de GRIHO Notícies (les darreres, amb enllaç a la pàg. de notícies) Esdeveniments (els darrers, amb enllaç a la pàg. d'esdeveniments) Links per tasques | <ul style="list-style-type: none"> ESTRUCTURA DEL LLOC WEB GRIHO: <p>PRESENTACIÓ Presentació Història</p> <p>MEMBRES El fundador El director Investigadors Col·laboradors Ex-col·laboradors</p> <p>LÍNIES DE RECERCA</p> <p>PROJECTES Projectes de recerca Projectes de transferència de tecnologia</p> <p>INFRAESTRUCTURES Usabilitat</p> <p>PUBLICACIONS Llibres Edicions Capítols Articles Reports de recerca</p> <p>DOCENCIA TFC Tesis</p> <p>LINKS RELLEVANTS Altres grups de recerca Congressos Revistes</p> <p>NOTÍCIES ESDEVENIMENTS: Seminaris, Congressos</p> |

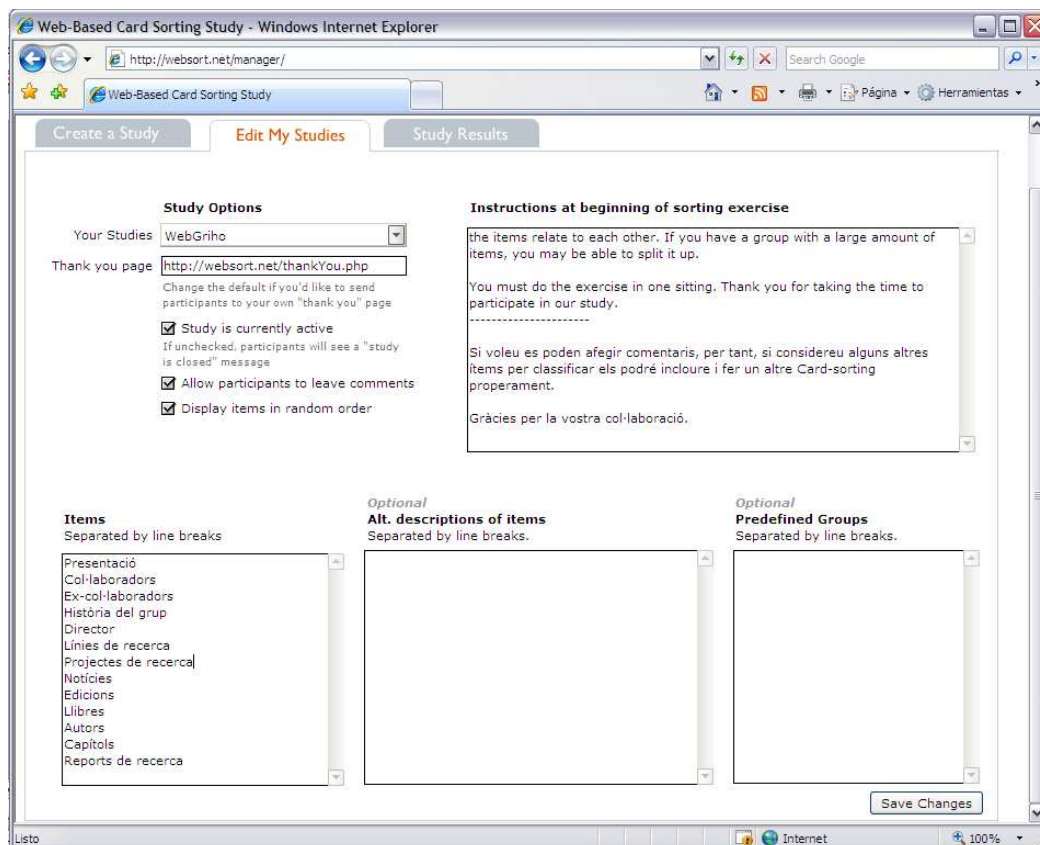
Per a la generació de l'estructura de la informació s'ha utilitzat la tècnica anomenada "CardSorting" o Ordenació de Targetes.

Amb el CardSorting es pot entendre com els usuaris imaginem la organització de la informació, és a dir, com agrupem els conceptes, per tant, amb aquesta tècnica s'inicia la construcció de l'arquitectura dels continguts de la web.

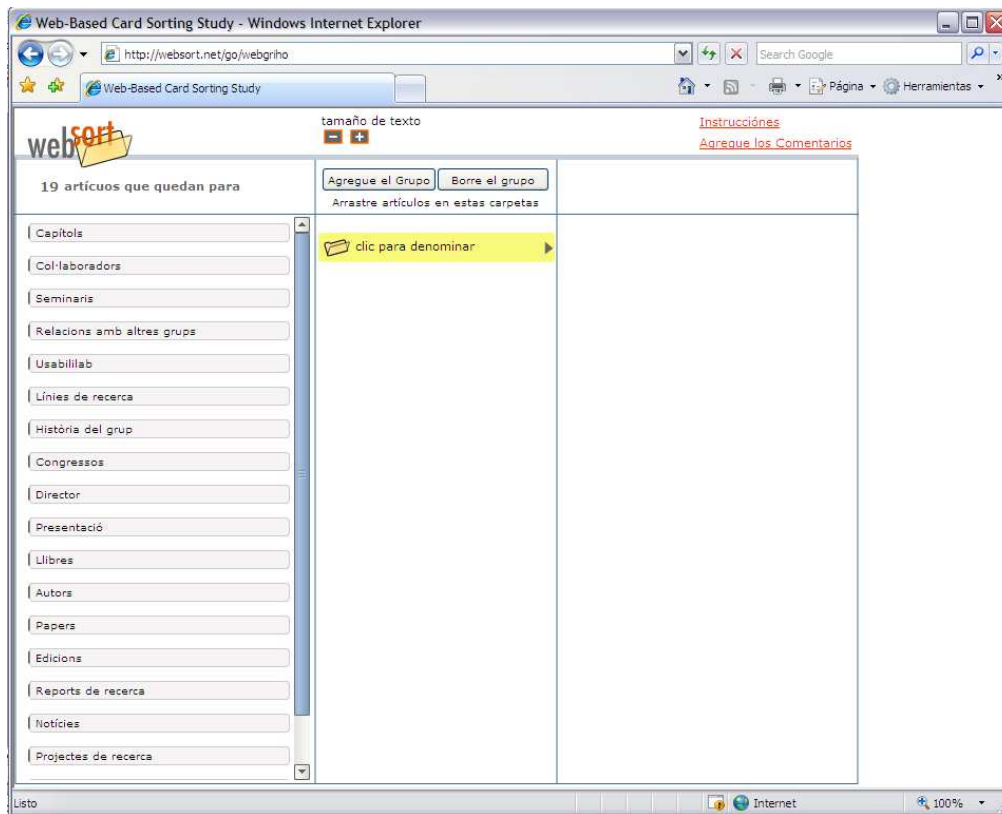
S'han utilitzat dues eines de CardSorting: la primera, el WebSort (<http://websort.net>), una eina basada en web que permet fer estudis remots d'ordenació de targetes, i la segona eina, el Card Sorting Clustering, desenvolupada en el TFC d'Enginyeria en Informàtica de Daniel Pardell Mateu, similar als CardZort i CardCluster de Jorge A. Toro (<http://www.cardzort.com/cardzort/index.htm>).

El procediment ha estat el següent:

1. Es crea un nou estudi amb el WebSort anomenat WebGriho i s'hi introdueixen tots els ítems que fan referència a cadascun dels continguts que es volen introduir a la nova web.



2. S'envia a tots els membres de Griho que participaran en l'estudi l'adreça a través de la qual podran fer l'ordenació dels ítems (<http://websort.net/go/WebGriho>), en la qual hi participen vuit membres del grup.



3. Una vegada tothom ha fet la classificació de la forma que ha cregut adequada es recullen els resultats, obtenint així totes les categories proposades i els ítems que han de pertànyer a cadascuna, juntament amb una gràfica en forma d'arbre que ajudarà a fer l'anàlisi.

CATEGORIES PROPOSADES RESULTANT DEL CARD SORTING:

INICI / PRESENTACIÓ / HISTÒRIA / QUI SOM? / QUÈ ÉS GRIHO?

MEMBRES / GRUP / EQUIP

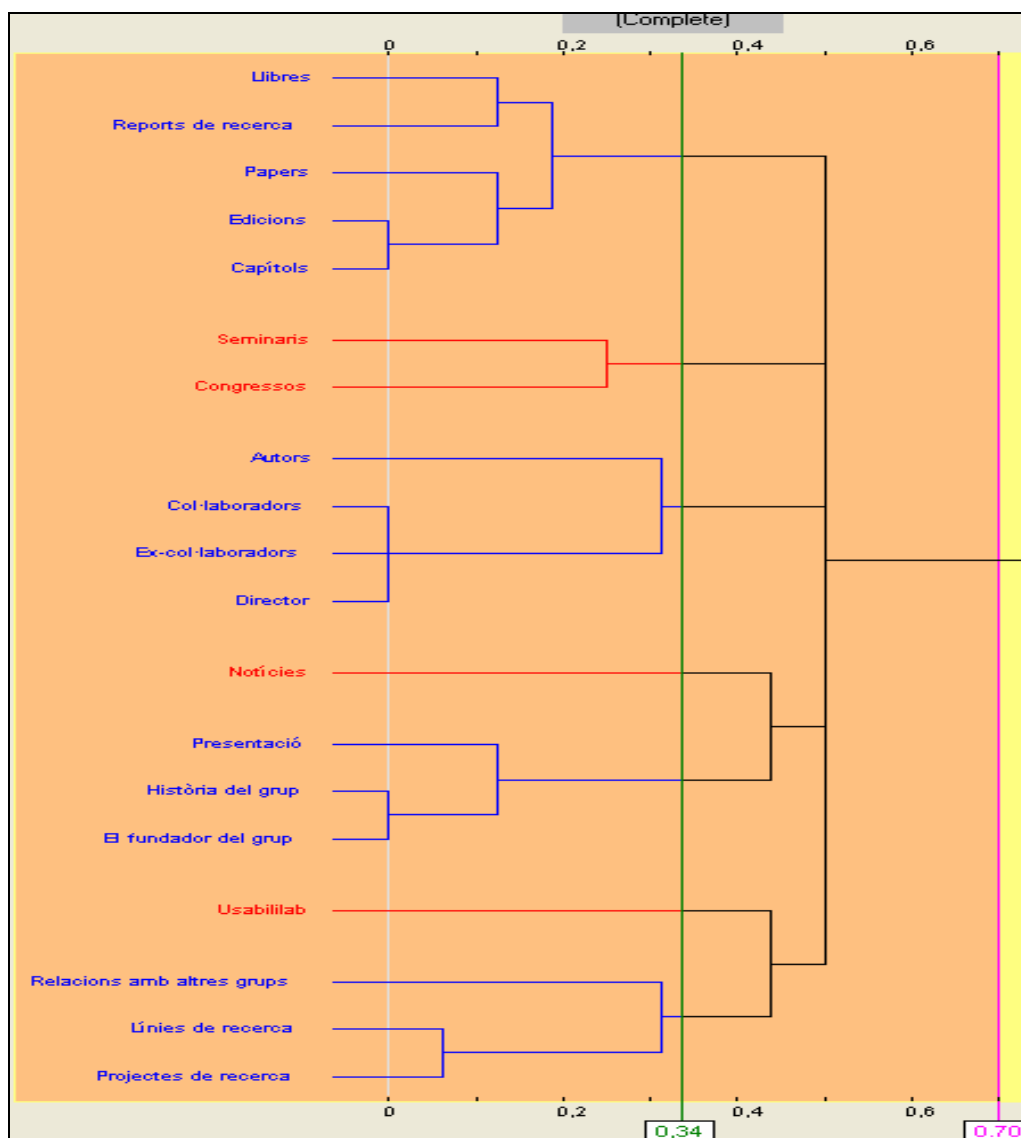
RECERCA / PROJECTES

PUBLICACIONS / RECURSOS

EQUIPAMENTS / INFRAESTRUCTURES / LABORATORI / USABILILAB

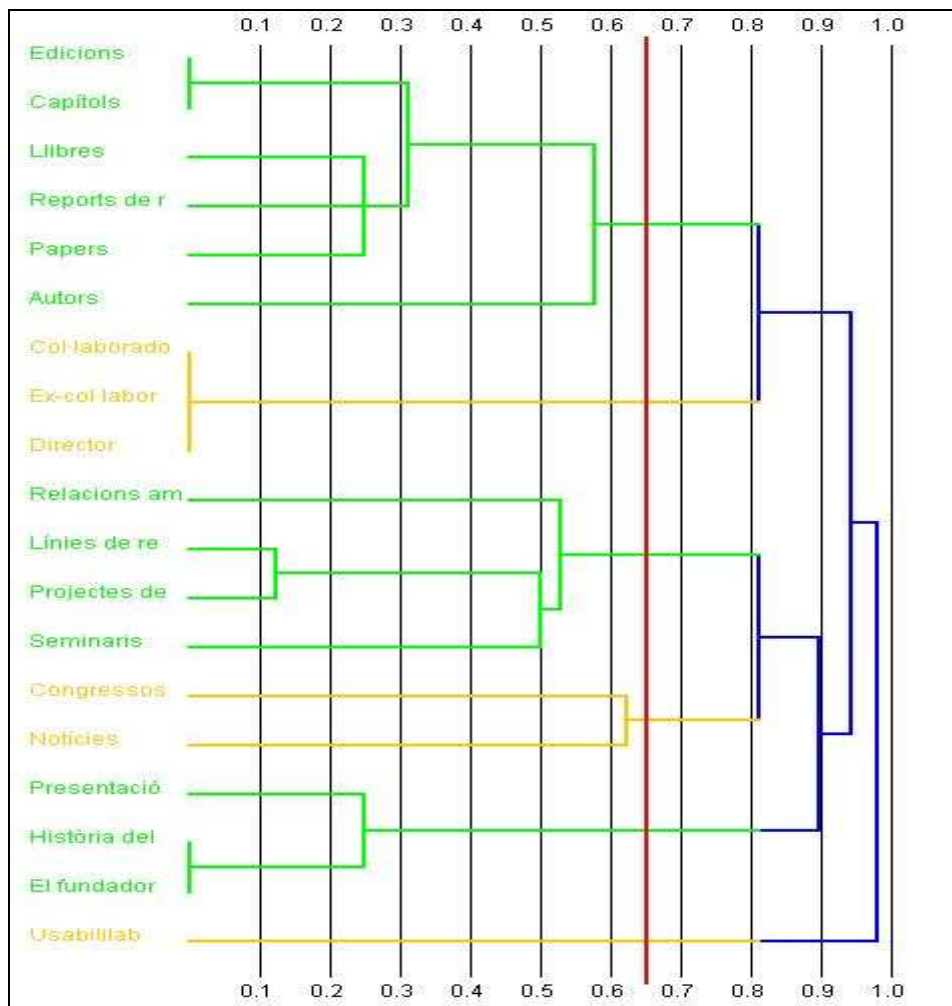
ACTIVITATS / DIFUSIÓ / EVENTS

NOTÍCIES / ACTUALITAT



4. En aquest punt és on s'utilitza l'altra eina de CardSorting per tal de poder fer una comparativa de resultats.

Per evitar que els implicats hagin de tornar a introduir la informació, i per evitar diferències amb la informació del test anterior, s'ha introduït directament tota la informació del primer test i s'ha generat un nou arbre de resultats.



5. Finalment, amb totes les categories proposades i els dos arbres de resultats es fa un nou Focus Group amb els membres de Griho que serveix per avaluar la informació recollida. La informació es representa en el quadre següent:

| <u>Resultat focus group</u> | <u>Resultat conf-confer</u> |
|------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|
| PRESENTACIÓ Presentació Història | PRESENTACIÓ Presentació Història Història El fundador del grup |
| MEMBRES El fundador El director Investigadors Col·laboradors Ex-col·laboradors | MEMBRES El director Investigadors Col·laboradors Ex-col·laboradors |
| LÍNIES DE RECERCA | RECERCA |
| PROJECTES Projectes de recerca Projectes de transferència de tecnologia | Línies de recerca Projectes Relacions amb altres grups Seminària |
| PUBLICACIONS Llibres Edicions Capítols Articles Reports de recerca | PUBLICACIONS Llibres Edicions Capítols Articles Reports de recerca |
| DOCÈNCIA TFC Tesis | INFRAESTRUCTURES Usabilitat |
| INFRAESTRUCTURES Usabilitat | ESDEVENIMENTS Congressos |
| LINKS RELLEVANTS Altres grups de recerca Congressos Revistes | NOTÍCIES Notícies |
| NOTÍCIES | |
| ESDEVENIMENTS : Seminària, Congressos | |

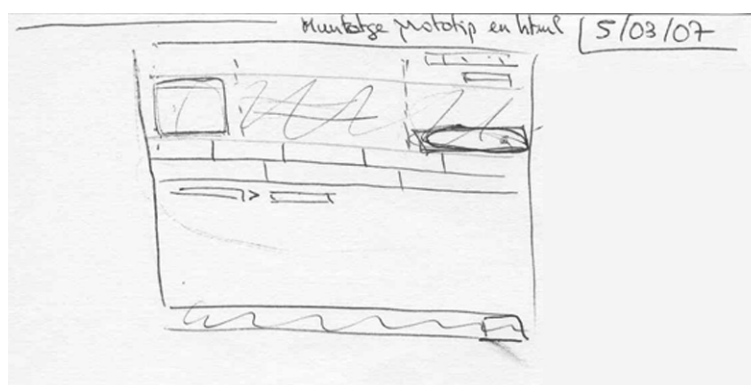
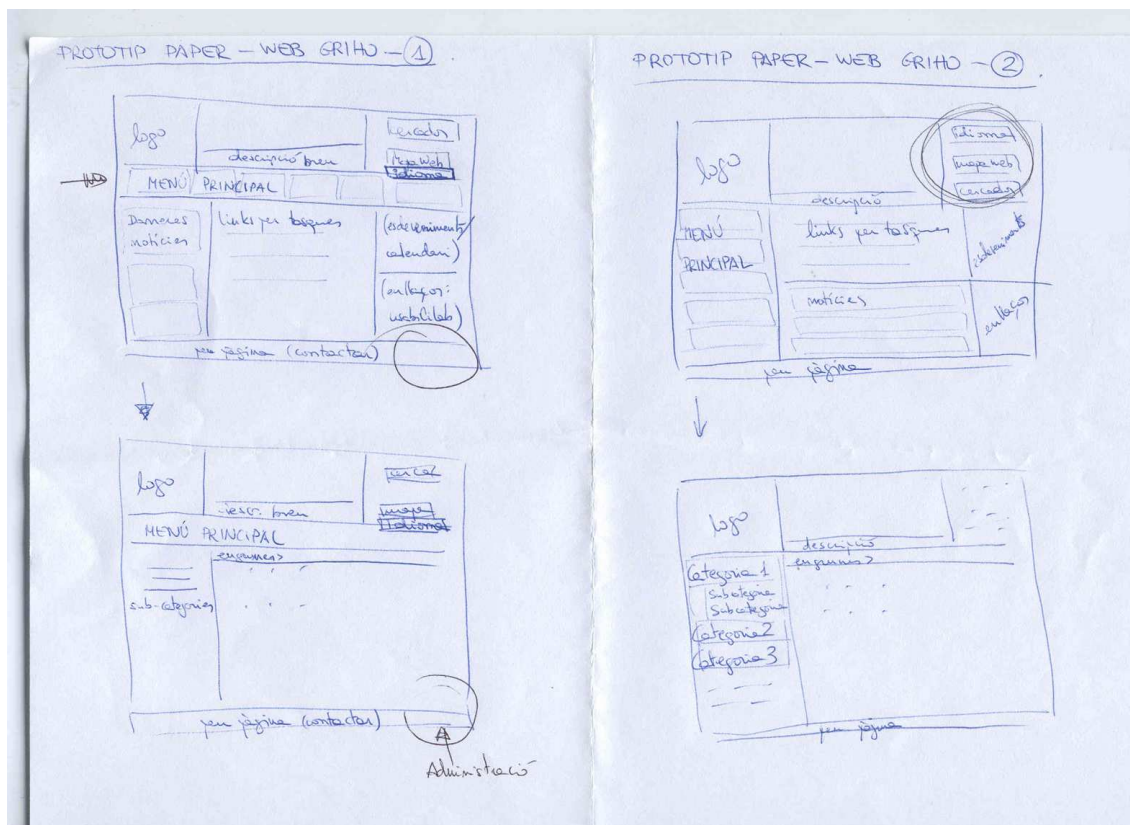
I així finalment es pot establir l'arquitectura definitiva de la informació de la nova web:

| | |
|-----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PRESENTACIÓ Presentació Historia | Pàgina inicial: <ul style="list-style-type: none">• Darreres notícies i esdeveniments• Links per tasques• Breu descripció de GRIHO• (calendar)??? |
| NOTÍCIES / ESDEVENIMENTS | |
| MEMBRES El fundador El director Investigadors Col·laboradors Ex-col·laboradors | Altres consideracions: <ul style="list-style-type: none">• Membres – foto i breu descripció + link opcional a la pag. personal• Publicacions – classificació per categories, any, autors + cercador |
| RECERCA Línies de recerca Projectes Tesis Relacions amb altres grups Congrésos rellevants Revistes | |
| PUBLICACIONS Llibres Edicions Capítols Articles Reports de recerca | |
| TRANSFERÈNCIA DE TECNOLOGIA Projectes CEUA | |
| DO CÈNCIA / FORMACIÓ Titulacions / Assignatures TFC Seminaris | |
| INFRAESTRUCTURES USABILLAB GRIHO | |

4.3. Prototipat

Amb els requisits del nou lloc web establerts i la informació organitzada amb una estructura concreta és el moment d'iniciar l'etapa de prototipat. En aquesta, es proposaran diverses opcions per al disseny de la interfície del lloc web.

El primer pas consisteix en crear diversos prototips de paper senzills per establir les posicions dels controls i dels elements de navegació i, en definitiva, per decidir quina serà l'estructura bàsica de les pàgines de la web.

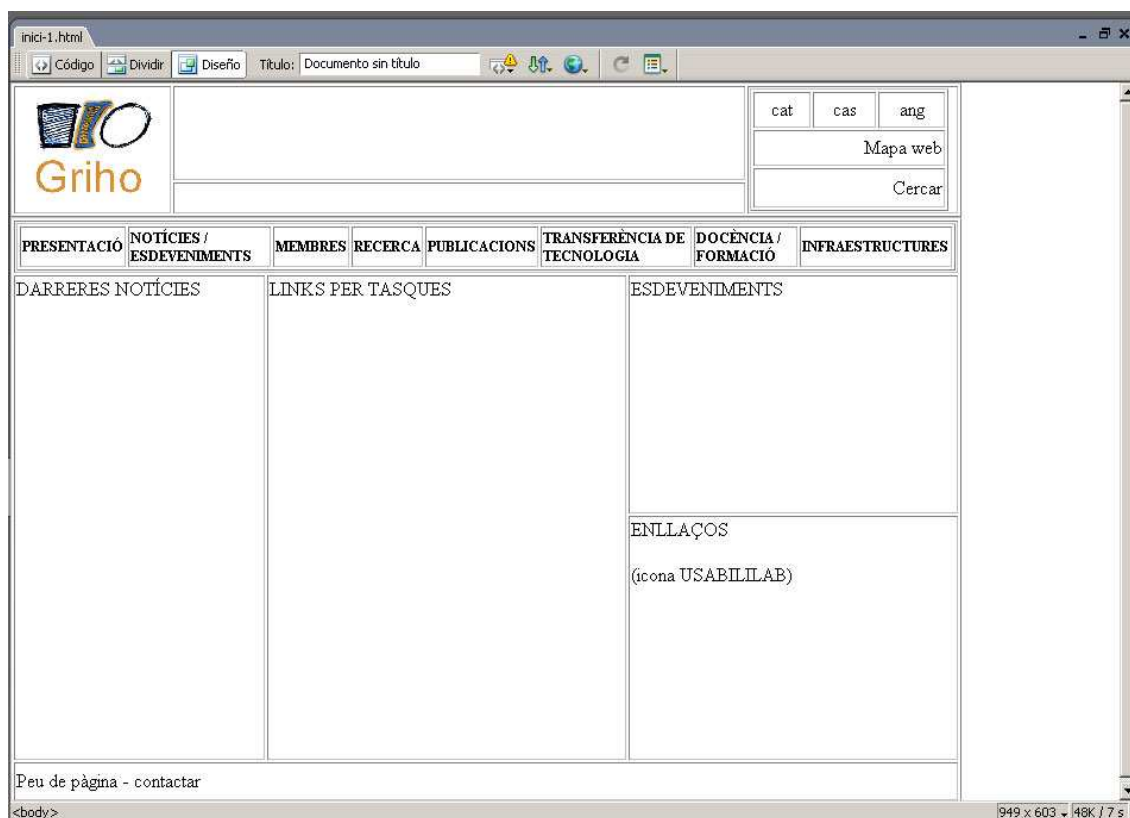


Una vegada s'ha escollit el model que s'adapta millor a l'arquitectura de la informació que tindrà la web i que mostra tots els elements de la interfície necessaris, a través de fer un altre Focus Group amb els membres de Griho, es procedeix a la creació d'un primer prototip de software.

Per fer els prototips de software s'utilitza el Macromedia Dreamweaver, ja que és una eina que facilita la creació de documents HTML d'ús molt comú i que ens permetrà treballar amb comoditat i fer els canvis necessaris de forma relativament senzilla.

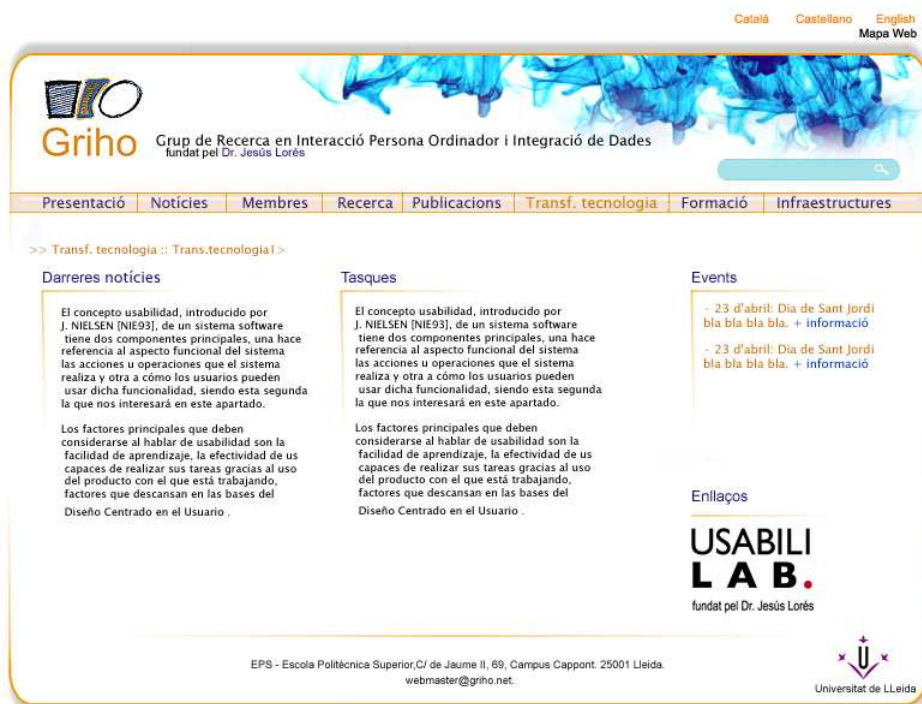
Cadascun dels prototips de software realitzats consta de dues pàgines; una que es correspon a la pàgina inicial del lloc web, i una altra representativa de les pàgines que contenen informació dins d'algun dels menús del lloc web.

Es comença creant un document HTML per cadascuna de les pàgines, generant només una estructura bàsica en forma de taula que representi l'estructura on s'encabiran els elements d'aquestes pàgines, per poder disposar de la seva “carcassa” bàsica.



En aquest punt és on es necessita aportar un disseny per a la interfície, per començar a definir l'aspecte general de les pàgines (formes, colors, imatges, ...).

Aquest disseny és elaborat per la Rosa Gil Iranzo, membre de Griho, que ha col·laborat extensament en tot el disseny de la interfície del lloc web i a qui agraeixo tot l'interès i dedicació que ha destinat a aquest projecte.



El procediment per aplicar el disseny al prototip inicial de software consisteix en dividir en imatges el disseny de la interfície i aplicar cadascuna d'aquestes imatges a la zona corresponent de la pàgina, havent separat les zones amb les etiquetes `<div>` d'HTML i indicant la seva posició i característiques en el full d'estil.

detall del codi de la pàgina inicial del prototip:

```
<html>
<head>
...
<link href="prototip_v1.css" rel="stylesheet" type="text/css" />
</head>
<body>
<div id="pagina">
  <div id="capsalera">
    
  </div>
  ...
</body>
```

detall del codi del full d'estil:

```
#capsalera {
  margin: 0;
  padding: 0;
}
```

Seguidament es van subsituïnt totes les imatges pels elements de text corresponents en el fitxer HTML i es va actualitzant el full d'estil amb els tipus de lletra adequats i posant les imatges de fons fins aconseguir generar el full d'estil bàsic.

detall del full d'estil bàsic:

```
#capsalera-esquerra {  
    margin: 0;  
    padding: 0;  
    float: left;  
    width: 60%;  
    height: 127px;  
    background: url(capsalera_esquerra.jpg) top left no-repeat;  
}
```

Aquest full d'estil s'anirà ampliant en funció de les necessitats i servirà posteriorment en la web final dins d'OpenCms.

Gràcies a la utilització del full d'estil es construeixen les pàgines líquides, fet que permet que el contingut de la pàgina s'adapti a la mida del navegador, en concret, a l'amplada de la finestra; així es pot visualitzar la mateixa pàgina en diferents resolucions de pantalla i en diferents mides de la finestra de navegador.

Cal considerar, però, que degut al disseny de la web, en especial, el disseny dels menús de navegació, la resolució mínima per veure les pàgines de manera òptima és 800x600 píxels, ja que en resolucions menors els elements del menú es desplacen i es fan difícils d'interpretar.

4.4. Implementació

La fase d'implementació és la penúltima del model de procés.

La implementació consisteix en escriure tot el codi que, aplicat sobre la plataforma escollida, és a dir, el gestor de continguts OpenCms, proporcionarà un sistema que compleixi amb les especificacions establertes en la fase d'anàlisi de requisits i d'acord amb les característiques del prototipat i de les avaluacions realitzades.

Així, després de seguir totes les fases prèvies del Model de Procés de l'Enginyeria de la Usabilitat i l'Accessibilitat (MPIu+a), i d'haver estudiat i analitzat les característiques i el funcionament d'OpenCms, es procedeix a iniciar la implementació del lloc web dissenyat.

Tots els detalls s'expliquen en els apartats següents, dedicant cada apartat a una etapa del procés d'implementació per poder tenir una visió més precisa i entenedora dels passos que s'han seguit.

4.4.1. Creació de l'espai per al lloc web dins d'OpenCms

Tal com s'indica al capítol sobre OpenCms, la primera fase per a la creació d'un nou lloc web consisteix en establir els espais que necessitarà la web dins d'OpenCms per poder procedir al seu desenvolupament.

Dins de l'espai de fitxers d'OpenCms, és a dir, dins de l'VFS, es crea un nou microsite que contindrà totes les pàgines de la web.

Seguidament, es crea un mòdul on s'hi emmagatzemaran totes les plantilles, recursos i elements; en definitiva, totes les dades que serveixen per donar format i funcionalitats a les pàgines de la web.

Així, en aquest punt, la base del lloc web que es començarà a construir té la següent estructura dins de l'VFS d'OpenCms:

- `/sites/default/microsite_griho`

És el directori on es podran trobar totes les pàgines del lloc web, és a dir, els fitxers HTML que contindran els continguts.

- `/system/modules/modul_griho`

És el directori que conté els subdirectoris:

- o `/templates`

On s'emmagatzemaran les plantilles utilitzades per les pàgines de la web.

- o `/resources`

On hi haurà tots els recursos que utilitzin les pàgines i les plantilles, com seran el full d'estil i les imatges, entre d'altres.

- o /elements

On s'aniran posant tots els fitxers JSP, o elements externs cridats des de les plantilles, que seran reutilitzats per aquestes. Serà el cas dels codis per construir la navegació o les engrunes de les pàgines.

4.4.2. Creació dels primers elements bàsics

Partint de les dades del primer prototip de software i un cop establerts tots els directoris que allotjaran els fitxers de la web es comença la implementació.

Es crea un primera plantilla dins de la carpeta “/templates”, el contingut de la qual és el mateix codi HTML d'una pàgina escollida del prototip de software, la pàgina inicial. Seguidament, dins del directori de recursos “/resources” es crea una nova carpeta anomenada “css” el contingut de la qual serà un fitxer de text amb extensió CSS i un nou directori anomenat “imatges”.

El fitxer de text és el full d'estil, i a dins del directori “imatges” s'hi posen tots els fitxers JPG que utilitza el full d'estil per donar l'aparença desitjada.

Com que l'estructura de directoris en el moment del desenvolupament del prototip de software era diferent de l'estructura utilitzada dins d'OpenCms, s'haurà de revisar tot el full d'estil i escriure correctament totes les rutes cap a les imatges.

Uns altres canvis que seran necessaris en el codi de la plantilla, respecte del prototip inicial, són la necessitat de modificar el “head” per aconseguir que la ruta cap al full d'estil recorri correctament l'estructura de directoris d'OpenCms. S'han de seguir les indicacions de l'apartat “L'ús de recursos a les plantilles” del capítol d'introducció a OpenCms.

En aquest punt de la implementació de la web es comencen a aplicar funcions pròpies d'OpenCms; s'ha d'utilitzar l'etiqueta `<cms:link>` de la llibreria de tags d'OpenCms.

Per a poder-ne fer ús, cal escriure a la primera línia del codi de la plantilla:

```
<%@ taglib prefix="cms" uri="http://www.opencms.org/taglib/cms" %>
```

per tal d'indicar al servidor que s'utilitzaran les etiquetes de la llibreria d'OpenCms.

Ara ja es pot canviar la línia que indica el full d'estil, posant-hi:

```
<link                                type="text/css"                                rel="stylesheet"
href="<cms:link>../resources/css/griho.css</cms:link>">
```

L'etiqueta `<cms:link>` construeix una URL vàlida a OpenCms per a qualsevol element que pertanyi a l'VFS; s'encarrega de generar la ruta completa cap al recurs.

Com que ara la plantilla ja està ben construïda, es pot passar al directori on hi haurà els continguts de la web, les seves pàgines, que és el directori corresponent al microsite.

Dins del microsite es crea un document HTML, una nova pàgina, i a la propietat que fa referència a la plantilla d'aquesta pàgina, la propietat "Template", s'hi selecciona la plantilla que hem creat.

No caldrà donar cap mena de continguts a la pàgina, ja que tota la informació ja està a la plantilla; així, quan es finalitza la creació del document i es visualitza fent clic sobre el nom del fitxer, apareixerà una pàgina idèntica a la del prototip.

4.4.3. Optimització del codi

Tornant al prototip de software en el qual es basa la implementació, es pot veure que hi ha diverses parts que són comunes per a la majoria de pàgines.

És el cas de la capçalera o "`<head>`" del codi HTML, que d'ara en endavant anomenarem "head" per evitar confusions.

També es repeteix el codi de capçalera de les pàgines en si, considerant ara la paraula “capçalera” com la porció de codi que crea les pàgines des de la seva part superior visible fins al punt on apareix el menú.

I els codis del menú, ja que a totes les pàgines hi apareix, i del peu de pàgina, a més d’alguns altres elements que aniran apareixent.

Per aconseguir la optimització del codi de les plantilles, s’agafa cadascuna d’aquestes parts de codi repetides i es guarden dins fitxers JSP en el directori “/elements” del mòdul.

Amb aquests canvis, si cada plantilla diferent per a unes pàgines en concret utilitza el mateix codi per construir la capçalera, per exemple, no s’han d’escriure totes aquestes línies a la plantilla, sinó que amb l’etiqueta “include” d’OpenCms es cridarà a l’element corresponent.

Un exemple és la substitució del codi del head de les plantilles per:

```
<cms:include file="../../elements/head.jsp" />
```

Un cop optimitzat el codi de la plantilla inicial que s’ha creat, i en aquest punt de la fase d’implementació, els fitxers que tindrem dins del directori “/elements” del mòdul són:

- head.jsp: amb els codi dels <head> de les pàgines
- capsalera.jsp: amb el contingut de la part superior de les pàgines fins arribar al menú
- menu.jsp: amb una llista dels noms dels elements que componen el menú; noms obtinguts en els resultats del Card Sorting.
- peu.jsp: amb el contingut del peu de les pàgines

4.4.4. Gestió d'idiomes

Una de les avantatges que ofereix OpenCms com a sistema gestor de continguts és la facilitat per gestionar els idiomes dels continguts de les pàgines.

Tal com s'havia comentat en el capítol d'OpenCms, des dels editors de continguts WYSIWYG es pot fer la introducció dels continguts per a un element en concret en tots els idiomes pels quals estigui habilitat OpenCms.

Seguint els resultats de la fase d'anàlisi de requisits i amb el prototip de la web, es decideix que les pàgines hauran d'estar disponibles en tres idiomes: català, castellà i anglès. Per tant, anem a tractar com es farà aquesta gestió d'idiomes.

- **Organització de les pàgines**

Per aconseguir un lloc web multilingüe es poden utilitzar diverses estructures, però en el cas del lloc web de GRIHO se n'utilitza una de les més recomanades. Aquesta organització es basa en crear dins del microsite un directori per a cada idioma.

El nom de cadascun dels directoris serà el codi format per lletres minúscules que identifiquen els idiomes configurats a OpenCms.

Dins del directori corresponent a cada idioma hi haurà totes les pàgines de la web, seguint sempre la mateixa estructura.

El procés de creació d'una pàgina per a que estigui disponible en tots els idiomes consisteix en crear la pàgina en un dels directoris i, seguidament, fer una còpia germana, també anomenada "sibling", d'aquesta pàgina en tots els directoris dels altres idiomes.

Un "sibling" és la còpia d'un recurs, amb la característica especial que un recurs i tots els seus germans estan associats, de manera que quan se'n modifiqui un, també es farà la modificació de tots els que hi estiguin associats.

Amb aquesta possibilitat es facilita notablement el manteniment dels continguts de les pàgines.

- **Selecció d'idioma**

Amb les pàgines del lloc web ja estructurades per idiomes, cal establir com fer la selecció del directori adequat segons l'idioma que triï l'usuari per navegar per la web.

A la part superior de totes les pàgines, en l'espai de la capçalera, és des d'on l'usuari pot seleccionar l'idioma.



Així, el codi d'aquesta capçalera, situat en el fitxer “capsalera.jsp” del directori “/elements” del mòdul, fa la gestió dels idiomes amb les següents instruccions:

```
<ul>
<li><a href="<cms:link>/sites/default/microsite_griho/en/inici.html
      __locale=en</cms:link>">English</a></li>
<li><a href="<cms:link>/sites/default/microsite_griho/cas/inici.html
      __locale=cas</cms:link>">Castellano</a></li>
<li><a href="<cms:link>/sites/default/microsite_griho/ca/inici.html
      __locale=ca</cms:link>">Català</a></li>
</ul>
```

Quan l'usuari clica sobre un dels idiomes per veure la web en aquest idioma en concret, l'acció que es porta a terme consisteix en enllaçar cap a la pàgina inicial dins del directori de l'idioma escollit, passant com a paràmetre el valor de la propietat “__locale” per a aquesta pàgina.

El pas d'aquest paràmetre serveix per assignar la propietat a la pàgina si no estava assignada, o corregir-la en cas que fos errònia, i assegurar que la pàgina es carregui bé.

- **Control d'idioma en cada pàgina**

Una pàgina es construeix segons una plantilla, i cada plantilla es compon de diversos elements. Així, per tots els elements, especialment pels que generen el “head” de les pàgines, la navegació, o pels que carreguen continguts específics, s’ha de tenir en compte la propietat “locale”.

Totes les pàgines que componen el lloc web tenen la propietat “locale” assignada amb el mateix valor que identifica el directori arrel d’idioma on estan situades. Per exemple, totes les pàgines en català, que estan en el directori “ca” dins del microsite, tenen el “locale” assignat amb valor “ca”.

En tots els elements que generen les pàgines i que poden gestionar continguts multilingües, a la part inicial del seu codi, s'utilitzen les funcions d'OpenCms que permeten obtenir l'idioma del context en el que s'està treballant.

Aquesta acció serveix per poder construir la navegació cap a les altres pàgines seguint amb el mateix idioma.

Seguint amb l'exemple del fitxer “capsalera.jsp”, veiem com es porta el control de l'idioma:

```
<%
    // Create a JSP action element
    CmsJspActionElement cms = new CmsJspActionElement(pageContext,
        request, response);

    String locale= cms.getRequestContext().getLocale().toString();
%>
...
<a href="<cms:link>/sites/default/microsite_griho/
    <%
        out.println(locale);
    %>
    /mapa.html?__locale=
    <%
        out.println(locale);
    %>
    </cms:link>">Mapa web</a>
```

En aquestes línies de codi es pot veure com el primer que es fa és obtenir l'idioma del context actual i guardar-lo.

Seguidament, sempre que es vulgui fer un enllaç, s'utilitza el valor d'aquest idioma per fer la redirecció al directori correcte dins del microsite i per passar per paràmetre la propietat d'idioma a la pàgina de destinació mantenint el mateix valor.

Els continguts propis de cada pàgina que es visualitzaran seran els pertanyents a l'esquema XML de la pàgina per a l'idioma indicat a la propietat “locale” de la pàgina.

4.4.5. Creació de les pàgines

A partir de la informació recollida a la fase d'anàlisi de requisits, i que fa referència als continguts de les pàgines que formaran part del lloc web, i del disseny proposat i aplicat als prototips de software, apareix la necessitat de crear més plantilles, partint de la inicial, per als diferents models de pàgina.

- **Creació de plantilles amb una estructura concreta**

En base al codi de la primera plantilla, que es corresponia a la pàgina inicial per al lloc web del prototip de software, es creen noves plantilles per a les altres pàgines.

Les primeres modificacions es fan a la part de codi que conforma l'estructura de la part central de les pàgines, on hi apareix el propi contingut de la pàgina.

A la plantilla inicial, la part central consta de tres divisions verticals; per tant, es creen noves plantilles amb una i dues divisions, respectivament, que mantenen sense canvis la capçalera de la pàgina, el peu, i la navegació, i s'adapten a les propostes estructurals de les pàgines del prototip de software.

- **Nous elements per a les plantilles**

Just després de la creació de la plantilla inicial, s'havia procedit a la optimització del seu codi creant elements externs, és a dir, fitxers JSP que s'inclouen a la plantilla quan són necessaris, i que afavoreixen la reutilització de codi.

Els elements externs també serveixen per afegir noves funcionalitats mantenint la senzillesa del codi de la plantilla; així, segons els continguts que s'hagin d'introduir a les noves pàgines, es creen nous fitxers JSP que es cridaran des de les plantilles corresponents.

- **Elements editables**

Per les pàgines que han de mostrar continguts estàtics, és a dir, informació introduïda pels responsables de la web que no necessita cap tractament especial abans de poder ser visualitzada, no cal incloure cap fitxer JSP a la plantilla; s'utilitzen els elements editables.

Aquests elements editables, tal com s'havia introduït als capítols dedicats a OpenCms, són elements que poden ser accedits des de les plantilles amb l'etiqueta `<cms:include element="[nom_element]" editable="true" />` de la taglib d'OpenCms, i la seva informació es guarda al fitxer XML que té associat la pàgina que utilitza la plantilla que inclou l'element.

En aquest punt cal actualitzar el codi de l'element "head.jsp" que carrega el `<head>` de totes les pàgines, afegint-hi `<cms:editable/>` per permetre la inserció d'aquests elements editables.

A la web de GRIHO, els elements editables s'utilitzen al cos de les pàgines de presentació, i al de les pàgines dels membres fundador i director, per exemple.



- **Fitxers JSP i continguts estructurats**

Per altres pàgines com la de “Notícies i Esdeveniments” i per la secció de “Darreres Notícies” de la pàgina inicial, sí que és imprescindible tractar la informació abans de mostrar-la.

En aquests casos, apareix la necessitat de crear nous fitxers JSP per aconseguir el resultat desitjat, alhora que es comença a treballar amb una altra de les característiques especials d’OpenCms: els tipus predefinits de dades.

Tant les notícies com els esdeveniments són un tipus de recurs classificat com a contingut estructurat, que OpenCms ja té definits i proporciona mètodes per tractar-los.

Per a fer-ne ús, es crea un nou directori dins del microsite, al mateix nivell que els directoris pels idiomes; una nova carpeta anomenada “modules”.

Dins d’aquesta, s’hi posaran dos subdirectoris: “events” per emmagatzemar tots els recursos de tipus “event” on es posarà la informació relativa a cada esdeveniment, i un altre directori anomenat “news”, amb contingut semblant al de l’anterior però dedicat a les notícies.

4.4.6. Utilització dels recursos

Ara ja es poden crear els fitxers JSP per tractar aquest tipus especial de recursos.

El seu codi, a més de tenir en compte l'idioma del context, utilitzarà funcions de la taglib d'OpenCms que permet seleccionar tots els recursos d'un tipus en concret situats en un lloc determinat de l'VFS i fer el tractament de la informació que contenen.

- **Obtenció d'un conjunt de recursos**

Per poder mostrar un llistat de les notícies del lloc web, per exemple, primer cal obtenir totes les notícies, i es fa de la següent manera:

```
<cms:contentload collector="allInFolderDateReleasedDesc"
    param="/sites/default/microsite_griho/modules/news
        /news_${number}.html|news|0" editable="true">
```

```
... codi per tractar cadascun dels recursos
```

```
</cms:contentload>
```

S'utilitza l'etiqueta `<cms:contentload>`, que agafa tots els recursos ubicats a la ruta indicada al primer paràmetre i que tenen un nom que es correspongui amb el format indicat a la part final de la ruta, que siguin del tipus indicat al segon paràmetre (en aquest cas, tipus "news"), i n'agafa tants com indiqui el tercer i últim paràmetre.



En aquest cas, el tercer paràmetre és 0, cosa que indica que agafi tots els recursos.

També cal observar el valor de l'atribut "collector", que indica l'ordre concret de tractament dels recursos; en aquest cas, es tractarà primer l'últim que hagi estat introduït.

- **Tractament d'un element d'un recurs**

Un cop s'han obtingut els recursos, ja es poden tractar.

Les característiques de l'etiqueta `<contentload>` fan que les instruccions que apareixen entre l'inici `<cms:contentload>` i el final `</cms:contentload>` actuïn en forma de bucle per sobre de cadascun dels recursos obtinguts.

Així, aquestes instruccions actuaran sobre els elements dels recursos, tractant un recurs rere l'altre. Per tant, en la primera volta, s'actua sobre el primer recurs obtingut, i es poden agafar els seus elements.

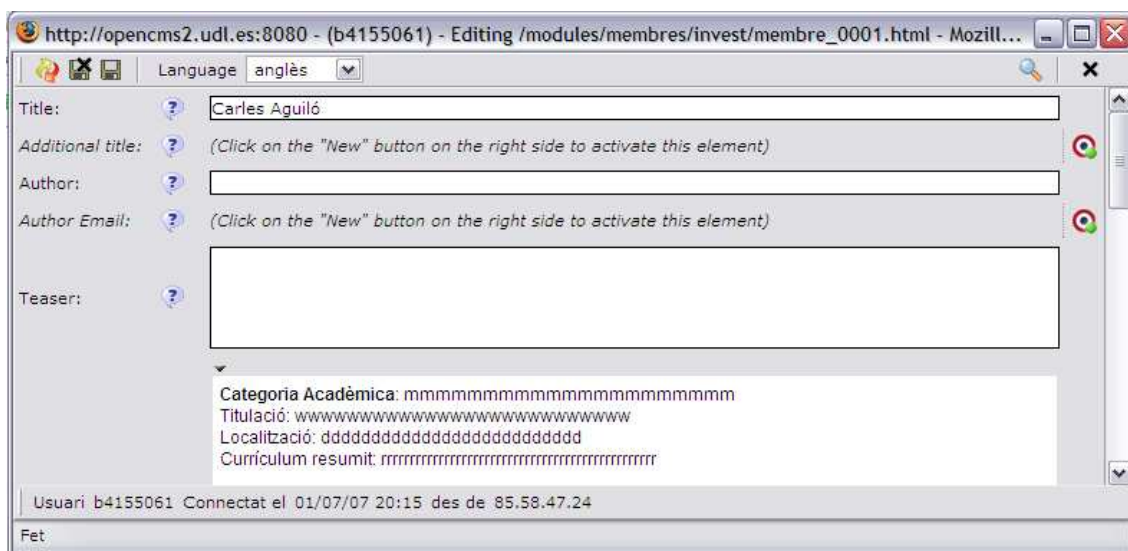
Això es fa amb l'etiqueta `<cms:contentshow>`, que permet mostrar la informació d'algun camp del recurs.

Per exemple, per mostrar el títol d'una notícia:

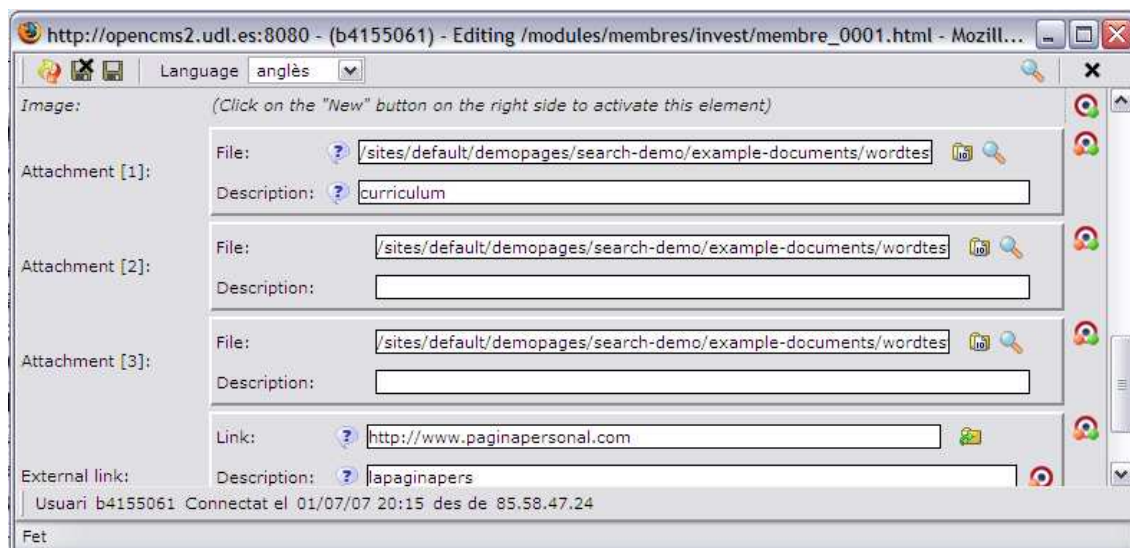
```
<cms:contentshow element="Title" />
```

Altres etiquetes útils per tractar elements dels recursos són `<cms:contentcheck>` i `<cms:contentloop>`.

La primera serveix per comprovar si algun camp en concret del recurs conté informació, i mostrar-la, per exemple, en el cas que sí que en contingui.



La segona serveix per recórrer els diversos elements d'un camp en concret que pugui tenir múltiples elements. Seria el cas de les imatges, els enllaços, o els arxius adjunts que poden tenir les notícies o els esdeveniments; amb el `<cms:contentloop>` els podrem recórrer tots i mostrar-los, si és el cas.



Un exemple on es fan servir aquestes etiquetes per mostrar els arxius adjunts d'un recurs, si és que en conté, i poder-hi enllaçar, és el següent:

```
<cms:contentcheck ifexists="Attachment">
  <ul>
    <cms:contentloop element="Attachment">
      <li><a href="<cms:link><cms:contentshow
        element="File" /></cms:link>" target="_blank"
        title="<cms:contentshow element="Description"
        />"><cms:contentshow element="Description"
        /></a></li>
    </cms:contentloop>
  </ul></td>
</cms:contentcheck>
```

Alguns camps dels recursos però, com les dates, requereixen un tractament especial, ja que es poden mostrar en diversos formats, per tant, s'han de formatgejar.

Per poder realitzar aquestes accions és necessària la utilització de la llibreria JSTL de Java.

- **Tractament especial d'un element d'un recurs**

La llibreria JSTL (JSP Standard Tag Library) és un component dins de l'especificació de J2EE format per un conjunt de llibreries d'etiquetes simples i estàndards que encapsulen la funcionalitat principal que serveix per escriure

pàgines JSP. Les etiquetes JSTL s'organitzen en quatre llibreries: core, xml, fmt i sql.

Als fitxers JSP que tractin els recursos s'haurà d'importar la llibreria “core”, que comprèn les funcions script bàsiques com bucles, condicionals i entrada/sortida, i la llibreria “fmt”, que comprèn la internacionalització i el format de valors de moneda i dates.

La importació d'aquestes llibreries s'aconsegueix afegint al començament del fitxer les línies:

```
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jstl/fmt" %>
```

I ja es pot tractar una data, per exemple, de la forma:

```
<c:set var="dateString"><cms:contentshow element="Date"/></c:set>
<%
    java.util.Date date = new java.util.Date();
    date.setTime(Long.parseLong(pageContext.getAttribute("date
String").toString()));
    pageContext.setAttribute("date", date);
%>
<fmt:formatDate value="${date}" type="date" pattern="dd/MM/yyyy"
/>
```

Primer, s'obté l'element “Date” amb l'etiqueta `<cms:contentshow>` del recurs que s'està tractant i es guarda dins d'una variable. Seguidament, dins del codi Java, es crea un nou objecte de tipus `Date` i se li assigna el valor de la variable obtinguda del recurs, prèviament passada a `Long`, ja que es tractava d'un `String`. A continuació, es posa la data en un nou atribut, i finalment, es dona el format a la data segons el patró indicat.

D'aquestes línies també en destaca la manera com s'ha de fer el tractament de les variables del context.

Per assignar valor a una variable amb JSTL, s'utilitza `<c:set>` i per obtenir-la o avaluar el seu valor, s'ha d'escriure `${[nom_variable]}`.

Per treballar amb aquestes variables dins de l'scriptlet JSP són necessàries les funcions d'OpenCms `getAttribute` i `setAttribute` sobre l'objecte `pageContext`, el context de la pàgina.

A més a més, la llibreria `fmt` permet l'assignació de l'idioma corresponent segons el context de visualització de les pàgines amb:

```
<fmt:setLocale value="${idioma}" />
```

i si la data es vol mostrar en format llarg i a la variable idioma s'hi ha posat el valor adequat, apareixerà la data en l'idioma correcte.

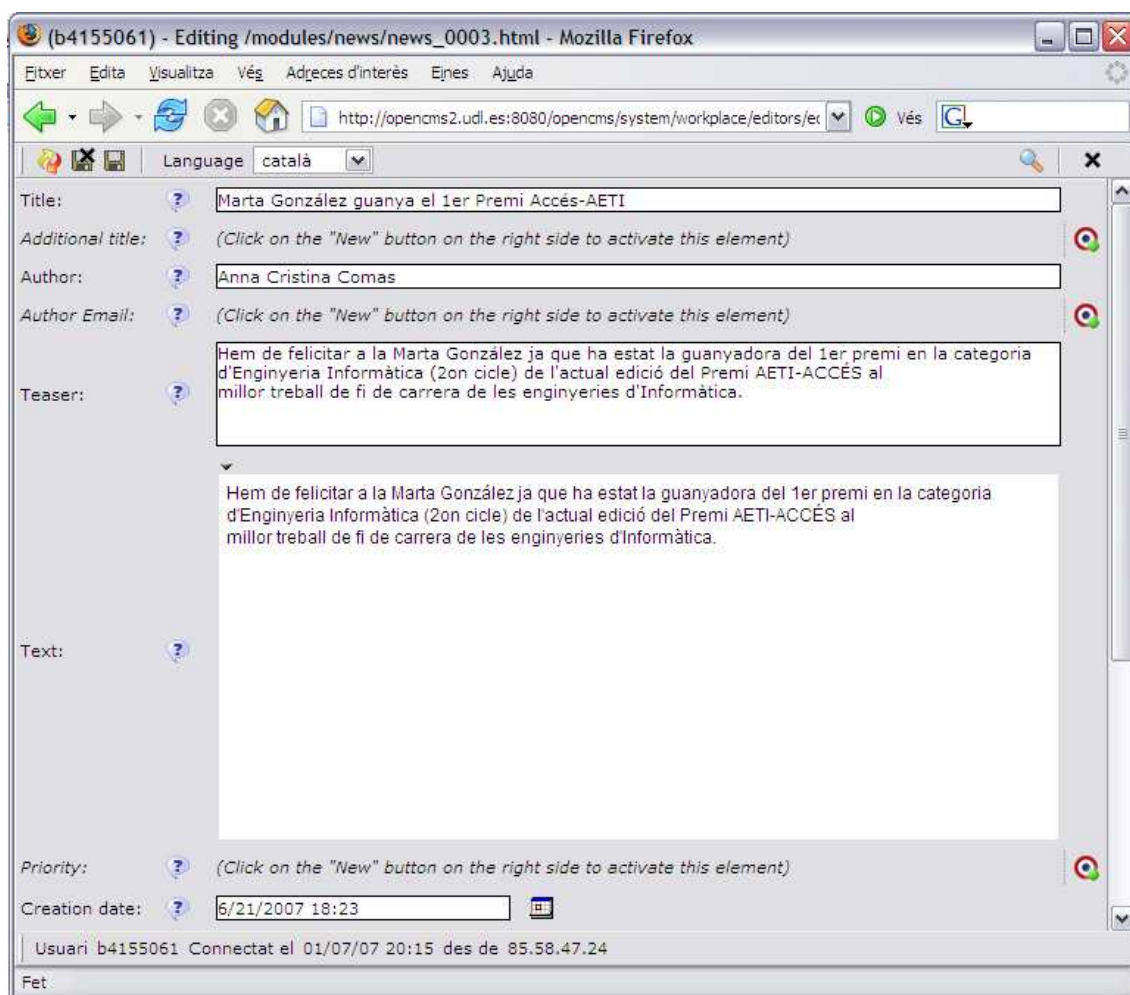
- **Obtenció d'un únic recurs**

L'etiqueta `<cms:contentload>` també serveix per agafar un sol recurs, enlloc de tot el conjunt.

Per exemple, si en el llistat de notícies apareix l'enllaç per a poder anar a la pàgina que conté tota la seva informació, construït de la forma:

```
<a href="<cms:link><cms:contentshow element="${opencms.filename}"  
/></cms:link>  
<%  
    out.print("?__locale="+locale);  
%>  
>"><cms:include element="info" /></a>
```

serà necessari associar a cadascuna de les pàgines de notícies una nova plantilla que importi un fitxer JSP que permeti tractar i mostrar tots els seus continguts.



La càrrega de continguts per a una única pàgina d'un recurs es fa amb:

```
<cms:contentload collector="singleFile" param="{opencms.uri}"
editable="true">

... codi per mostrar i tractar els elements del recurs

</cms:contentload>
```

On s'indica amb la variable `{opencms.uri}` que s'agafin els elements de la ruta indicada.

El tractament dels elements d'aquest recurs es farà tal i com s'ha descrit en els dos punts anteriors.

4.4.7. Creació dels elements de navegació

Amb la majoria de les pàgines del lloc web ja creades, és el moment de donar funcionalitat a la zona dedicada al menú, ja que fins ara es tractava només d'una simple llista.

També s'afegiran altres elements per millorar la navegació del lloc web com els submenús, les engrunes i una pàgina nova on aparegui el mapa web.

- **El menú i els submenús**

Els punts de menú que ha de tenir el lloc web per aconseguir una estructura entenedora i que faciliti la recerca de la informació que ofereix la web són els elements resultants del Card-Sorting, analitzats i acordats posteriorment amb els diversos Focus Grups.

La generació del menú i tots els submenús s'aconsegueix amb la utilització d'un tipus de recurs especial que no s'havia comentat fins ara. Es tracta dels continguts estructurats "linklist".

Un linklist és un recurs que disposa de diversos camps predefinits per emmagatzemar-hi informació, i de la mateixa manera que els altres recursos d'OpenCms, permet introduir els continguts en diferents idiomes.

Els camps que s'utilitzaran seran els que fan referència al títol i a la URL, i serviran per posar el nom de cada punt de menú i la ruta a la qual es troba la seva pàgina inicial, respectivament.



Així, tant per al menú, com per cadascun dels submenús, es crearà un directori dins la carpeta “modules” del microsite on hi ha la nostra web.

Seguidament, a dins de cada directori hi posarem tants fitxers linklist com categories tingui el menú o submenú.

A continuació, caldrà omplir els camps títol i URL amb les dades que han d'aparèixer a la navegació i la ruta cap a la pàgina inicial de cada secció, respectivament, i per cadascun dels idiomes disponibles.

Finalment, amb tota la informació introduïda i utilitzant l'etiqueta `<cms:contentload>` de la llibreria d'OpenCms en els fitxers JSP encarregats de construir aquests elements de navegació, aconseguirem generar el menú i tots els submenús del lloc web.

- **Les engrunes**

Les engrunes són un element de navegació que serveix per mostrar al visitant del lloc web la ruta fins a la informació actual que està consultant. Són molt útils perquè indiquen a l'usuari en tot moment on es troba dins de l'estructura de categories del lloc web.



Es tracta d'una senzilla línia de text que afavoreix notablement la usabilitat dels llocs web, tal com indica Jakob Nielsen, un dels impulsors de l'Enginyeria de la Usabilitat, en el seu article "Breadcrumb Navigation Increasingly Useful" (<http://www.useit.com/alertbox/breadcrumbs.html>).

La generació de les engrunes es basa en l'estructura de directoris en la qual s'organitzen les pàgines.

És per aquest motiu, que durant la creació de les pàgines del lloc web s'ha tingut molt en compte la ubicació de cadascuna d'elles.

A continuació s'explica el funcionament de les engrunes en el lloc web i les consideracions que caldrà tenir en compte durant les tasques de manteniment de la web, detalls que també es tractaran a l'annex adjuntat amb aquest treball.

En el cas es vulgui afegir alguna pàgina nova, el fet de seguir aquest patró serà imprescindible per aconseguir la generació correcta d'aquest element de navegació.

L'estructura és la següent:

Les pàgines que s'han d'enllaçar des del menú principal i que no necessiten submenús, es situen directament dins de la carpeta arrel de cada idioma.

Les pàgines que s'han d'enllaçar des del menú principal i que tenen submenús, es posen dins d'una carpeta, creant una carpeta nova per cada grup de pàgines pertanyents a un punt de menú.

I dins d'aquest segon grup de pàgines, si es tracta de la pàgina principal del submenú, o bé de la única pàgina que pot ser accedida des del punt de submenú, es posa directament a la carpeta creada; sinó serà necessari crear un subdirectori per cada punt de submenú que contindrà les seves pàgines.

Totes aquestes carpetes han de mantenir el mateix nom en els diferents idiomes, variant-ne només els títols; cada títol en l'idioma corresponent, ja que serà el contingut que es visualitzarà.

En el procés de creació de les engrunes però, s'han de tenir en compte alguns casos especials.

Ja hem vist que algunes de les pàgines, les que són d'algun dels tipus de contingut estructurat, és a dir, les notícies i els esdeveniments, estan ubicades a la carpeta “modules” del microsite.

Les engrunes per aquestes pàgines no es construeixen seguint l'ordre de directoris, sinó que en el codi s'introdueix directament l'estructura lògica de directoris.

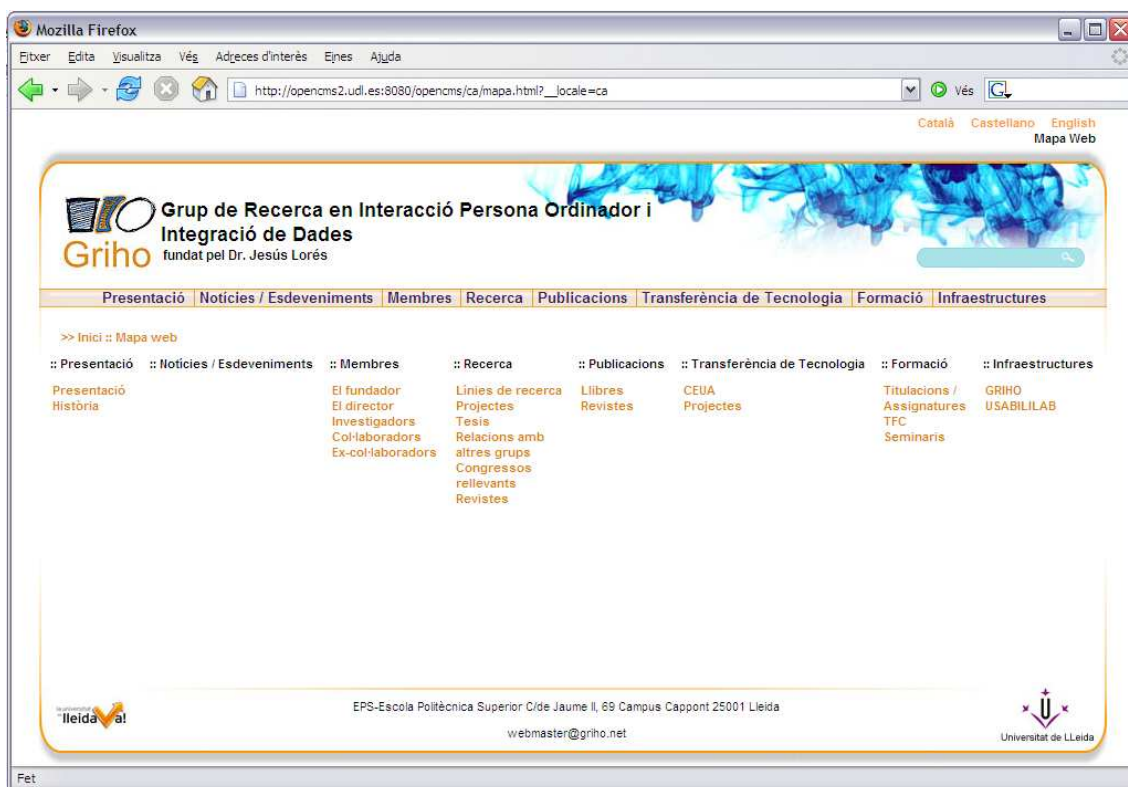
Per entendre-ho millor, agafem un exemple:

Suposant una notícia, la pàgina de la qual és “news_0001.html” i situada dins de l'VFS en els directoris “/sites/default/microsite_griho/modules/news”.

L'estructura de les engrunes per a aquesta pàgina, seguint l'estructura lògica del lloc web, haurà de ser “>> Inici >> Notícies / Esdeveniments :: [títol_de_la_notícia]“, per tant, aquesta navegació es generarà en un fitxer JSP especial per a aquestes pàgines.

- **El mapa web**

És el darrer element de navegació. Es tracta d'una pàgina convencional, la plantilla de la qual té una única divisió a la seva part central, on s'hi carrega un fitxer JSP que aprofita els codis de generació del menú i els submenús per obtenir el contingut.



4.4.8. Introducció de noves funcionalitats

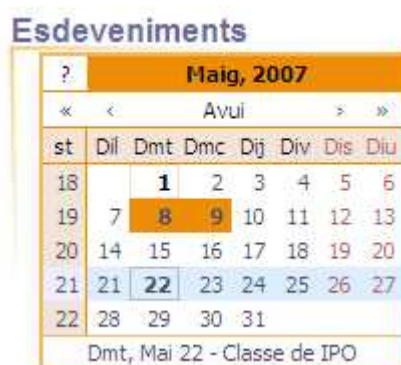
Amb la gran majoria del lloc web ja construït i pràcticament totes les pàgines creades, es decideix afegir noves funcionalitats en alguns punts de lloc web i implementar-ne d'altres, com el cercador, que ja estava previst. Són els temes que es tracten en aquest apartat.

- **Calendari d'esdeveniments**

En el prototip de software s'hi preveia que a la pàgina d'inici hi aparegués un espai dedicat als esdeveniments més rellevants.

En aquest moment de la implementació, sorgeix la possibilitat de substituir aquest llistat d'esdeveniments per un calendari que ressalti les dates amb esdeveniments.

Així, tots els esdeveniments que hagin estat prèviament introduïts pels administradors de la web es podran veure i ser accedits a través del calendari.



El sistema que s'utilitza és un calendari que permet marcar dates especials; es tracta del DHTML Calendar Widget de Dynarch, disponible a <http://www.dynarch.com/demos/jscalendar/>.

Aquest calendari està compost de diversos fitxers. Primer, un fitxer JavaScript amb el programa principal que defineix tota la lògica del calendari, diverses fulles d'estil o fitxers CSS, de les quals se n'ha escollit una que posteriorment s'ha adaptat especialment per al lloc web de Griho, i finalment disposa dels fitxers de definició dels llenguatges, que són fitxers JavaScript amb els textos que mostra el calendari.

Com que el calendari és un nou recurs per al nostre lloc web, els seus fitxers aniran dins d'un nou directori anomenat "js" de la carpeta de recursos (/resources) del mòdul de la web.

En el template de la pàgina inicial, s'ha d'incloure un nou element, el fitxer “`calendari.jsp`”, la funció del qual és recórrer tots els esdeveniments per guardar la seva data i el títol, per poder carregar aquestes dates especials en generar el calendari i fer que el calendari aparegui correctament.

Com que en aquest fitxer també es fa la posada en marxa del calendari, hi apareixen les instruccions necessàries per accionar el procediment adequat si el clica sobre una data especial.

Tant si en la data seleccionada només hi ha un esdeveniment, com si n'hi ha més d'un, s'enllaçarà automàticament a la pàgina que contindrà la informació desitjada.

Cal remarcar que la incorporació del calendari a la web del grup de recerca GRIHO ha estat un dels punts que ha suposat més dificultat en la implementació. El fet d'afegir aquesta funcionalitat ha requerit molt temps de recerca, ja que per encabir un element d'aquest tipus en el sistema OpenCms hauria sigut desitjable que es tractés d'un calendari escrit en JSP però va ser impossible trobar-ne amb codi lliure i per tant, aconseguir-ne un que s'adaptés amb facilitat va suposar un gran esforç. A més d'haver d'aconseguir un llenguatge compatible va ser necessari documentar-se a fons sobre el funcionament del calendari escollit per poder-lo enllaçar i fer-lo interactuar amb les nostres dades.

- **Cercador**

Per implementar el cercador de la web, s'utilitza el motor de cerca que ja porta incorporat OpenCms.

Aquest motor de cerca, igual que la majoria, llegeix d'una col·lecció de dades i crea un catàleg especial de totes les paraules que apareixen en aquestes dades. La informació que es guarda indica en quins documents es troben les paraules, en quina posició, o quantes vegades hi apareixen. És el que s'anomena un índex.

Per tant, caldrà crear un índex de cerca des de la vista d'administració d'OpenCms per a l'espai on es troben els continguts del lloc web, i tenint en

compte que en un índex de cerca només s'hi pot posar un idioma, se'n crearan tants com idiomes tingui el nostre lloc web.

Seguidament, s'ha de crear un “source” per al lloc web, també des de la vista d'administració, per indicar la ruta a partir de la qual s'han d'indexar els continguts, i quins tipus de continguts són, i, finalment, assignar-hi els índexs creats anteriorment.

The screenshot shows two panels from the OpenCms administration interface. The top panel, titled "Search Management", contains the following fields: "Name" with the value "Griho ca", "Rebuild mode" with the value "auto", "Locale" with the value "ca", and "Project" with the value "Online". The bottom panel, titled "Index sources", shows "Index sources (1)". It includes three buttons: "Show document types", "Show resources", and "Print". Below these buttons is a table with two columns: "Name" and "Indexer". The table contains one entry: "sourceGriho" in the "Name" column and "class org.opencms.search.CmsVfsIndexer" in the "Indexer" column.

| Name | Indexer |
|-------------|----------------------------------------|
| sourceGriho | class org.opencms.search.CmsVfsIndexer |

Amb els índexs de cerca preparats, ja es pot crear el fitxer JSP que doni la funcionalitat a l'espai dedicat al cercador que hi ha a la capçalera de les pàgines i la plantilla que s'associarà a la pàgina de resultats, per tractar i mostrar els resultats.

- **Gestió dels membres del grup de recerca**

Per a la gestió de la informació relativa als membres investigadors, col·laboradors i ex-col·laboradors del grup dins de l'apartat de Membres del lloc web de Griho s'aprofiten les possibilitats dels continguts estructurats que ofereix OpenCms.

Així, reaprofitant alguns dels camps del tipus estructurat “news” es pot facilitar la gestió de la informació de cada membre.

En el camp del títol s'hi posarà el nom del membre, a l'apartat de text, una breu descripció, i s'hi podran afegir arxius adjunts com el currículum vitae i un enllaç cap a la seva pàgina personal.

Amb aquest tipus de pàgines esdevindrà més senzill el manegament de la informació de cada membre i els continguts de les pàgines de membres de la web s'actualitzaran automàticament amb els continguts de cadascú, tal com passa amb els continguts de les notícies i dels esdeveniments.

- **Càrrega d'informació externa**

Finalment, per alguns continguts de la web, com és el cas de les publicacions del grup de recerca, s'ha optat per carregar dinàmicament la informació.

A partir d'una consulta a l'aplicatiu GREC es retorna automàticament aquesta informació actualitzada constantment s'evita que els administradors de la web de Griho hagin d'incloure nous continguts contínuament.

GREC és una aplicació de Gestió de la Recerca desenvolupada per la Universitat de Barcelona, actualment utilitzada per diverses institucions i organismes de recerca.

GREC inclou un conjunt de bases de dades (com per exemple, *Curricula Vitarum*, projectes, publicacions, etc...), uns entorns de suport (taules, informació, registre, ...) i uns programes d'explotació (informes, llistats, estadístiques, ...).

La seva adreça és <http://webgrec.udl.es/>.

Les dades retornades seran tractades i mostrades a les pàgines del lloc web.

Un inconvenient d'aquest sistema es troba en la impossibilitat d'indexar aquests continguts per a la cerca, ja que com que la informació no consta dins de les pàgines de l'VFS dedicades al lloc web, no apareixen als índexs ni sortiran als resultats.

4.4.9. Revisió del full d'estil

Un cop està tota la web creada, amb totes les pàgines i tots els seus continguts, i veient que tot funciona de la forma esperada, és el moment de la darrera revisió del full d'estil.

Per tots els elements que s'han anat afegint, com les notícies, els esdeveniments, les publicacions o els detalls dels membres del grup, i pel control de menús i submenús actius i el format dels enllaços, s'ha d'actualitzar el full d'estil per donar a tots aquests continguts l'aparença correcta.

És un darrer pas, bastant senzill, ja que durant la codificació de les plantilles i dels fitxers JSP s'havien estructurat totes les dades que havien d'aparèixer a les pàgines mitjançant capes (etiquetes `<div>` de l'HTML) associades al fitxer CSS mitjançant classes (`.[nom_classe]`) i identificadors (`#[nom_identificador]`).

4.5. Llançament

La fase de llançament és una de les més crítiques de tot el procés, ja que és el moment en el qual es veuen concretades en major o menor grau les expectatives posades en el producte.

Es pot indicar que l'èxit total del producte dependrà de dos factors molt importants: per un costat que l'usuari es senti còmode amb el sistema, és a dir, que no li doni errors i que no li resulti complicat recordar on estan les diferents opcions i les seves funcionalitats, i per altre costat que els responsables obtinguin els resultats esperats.

El fet d'haver desenvolupat aquest projecte en base al cicle de vida de l'Enginyeria de la Usabilitat assegura que ambdós aspectes es vegin satisfets.

El procediment que s'ha portat a terme en aquesta etapa ha consistit primer en el trasllat del lloc web, és a dir, de tots els fitxers que el componen, des del servidor local en el qual s'havia implementat cap al servidor de la UdL.

La tasca inicial ha consistit en revisar i modificar totes les rutes, les línies de codi, que indicaven el camí cap als diferents fitxers del sistema, ja que en la nova plataforma els noms dels directoris eren lleugerament diferents.

Després ha estat necessària la incorporació dels mòduls que contenen la informació relativa als tipus de dades estructurades, és a dir, les “news”, els “events” i els “linklists” que s’han descrit a l’apartat d’implementació, ja que es tracta de mòduls que es poden afegir opcionalment al servidor.

Havent fet aquestes adaptacions, es comprova que el funcionament del lloc web és totalment correcte; s’han complert els objectius funcionals.

Capítol 5

Conclusions i futures extensions

Amb el treball realitzat durant el desenvolupament d'aquest projecte s'ha aconseguit l'objectiu bàsic de redissenyar la web del grup de recerca GRIHO.

Gràcies a l'aplicació de les fases del Model de Procés de l'Enginyeria de la usabilitat i l'accessibilitat (MPIu+a) he pogut aprendre en profunditat les tècniques que hi intervenen i veure que la seva posada en pràctica garanteix l'èxit del sistema resultant, ja que en tot moment els membres del grup de recerca han pogut participar en el procés de desenvolupament de la web.

A més penso que un dels punts clau que han permès aquest bon resultat ha estat el gran esforç de treball en equip.

Per altra banda, des d'un punt de vista més dirigit a la implementació, altres conclusions que puc extreure són que a més d'aprendre a treballar amb llenguatges amb els quals no estava familiaritzada, com el JavaScript, he après molt en el desenvolupament de programari amb JSP.

Finalment, pel que fa a la utilització del sistema OpenCms, conèixer el concepte, les possibilitats i les funcionalitats que ofereix i aprendre-les a gestionar amb comoditat ha estat un dels factors que més valoro del meu treball en aquest projecte.

A partir de l'aprofundiment en el sistema gestor OpenCms, veig com a treball futur la possibilitat de creació de nous continguts estructurats que s'adaptin millor a les necessitats del grup.

Un element important és un tipus estructurat de dades que optimitzi el manegament de la informació dels membres del grup, definint una estructura específica, per als camps que es considerin necessaris, que sigui reconeguda pel sistema.

També seria interessant fer més proves d'usabilitat, sobretot a partir d'experimentar tasques reals en el sistema ja funcionant i/o avaluacions heurístiques que posin de manifest el treball realitzat, i comprovar el compliment dels objectius d'accessibilitat que s'havien plantejat amb les eines proporcionades per l'W3C.

Bibliografia

- Butcher, M. (2006). *Managing and Customizing OpenCms 6. A practical guide to creating and managing your own website with this proven Java/JSP-based content management system*. Packt Publishing Ltd.
- Granollers, T. (2005). *MPIu+a. Una metodología que integra la ingeniería del software, la integración persona-ordenador y la accesibilidad en el contexto de equipos de desarrollo multidisciplinares*. Tesis doctoral.
- Lacasa, A. (2006). *Exportación e Importación en OpenCms6*. Manual disponible a OpenCmsHispano (www.opencmshispano.org).
- Raposo, S. (2007). *Buscador de Contenido. Lucene*. Manual disponible a OpenCmsHispano (www.opencmshispano.org).

Referències web

- Pàgina web *Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad. MPIu+a*. Accessible a <http://griho.udl.es/mpiua> (04/07/2007).
- Pàgina web de l'W3C, el consorci encarregat de desenvolupar tecnologies per a portar la web al seu màxim potencial. Accessible a <http://www.w3.org> (04/07/2007).
- Article de Jakob Nielsen *Breadcrumb Navigation Increasingly Useful*. Accessible a: <http://www.useit.com/alertbox/breadcrumbs.html> (04/07/2007).
- Pàgina web d'OpenCms, on s'hi pot trobar el programari per a descarregar, juntament amb manuals i accessos a altres llocs de referència, entre altra informació. Accessible a <http://www.opencms.org> (04/07/2007).

- Pàgina web d'OpenCmsHispano, amb informació i documentació sobre l'ús d'OpenCms, tutorials i codi útil. Accessible a www.opencmshispano.org (10/06/2007).
- Fòrums de discussió sobre OpenCms amb informació per al desenvolupament i codi útil:
<http://www.opencms-forum.de> (04/07/2007)
<http://www.nabble.com/OpenCMS---Dev-f654.html> (04/07/2007)
<http://mail.opencms.org/pipermail/opencms-dev> (04/07/2007)
- Pàgina web *The Coolest DHTML / JavaScript Calendar* amb recursos de JavaScript. Accessible a <http://www.dynarch.com/projects/calendar> (04/07/2007)
- Pàgina web *Java™ 2 Platform, Standard Edition, v 1.4.2 API Specification*. Accessible a <http://java.sun.com/j2se/1.4.2/docs/api> (04/07/2007)
- Pàgina web *OpenCms Core API, version 6.2.3*. Accessible a <http://www.opencms.org/export/javadoc/core/overview-summary.html> (23/06/2007)
- Pàgina web per realitzar el Card-Sorting *WebSort*. Accessible a <http://www.websort.net> (04/07/2007)

Annex A

Guia d'administració dels continguts del lloc web GRIHO

Aquesta guia pretén recollir els passos bàsics i fonamentals que podran seguir les persones encarregades de la gestió dels continguts dels lloc web del grup de recerca GRIHO per a poder desenvolupar les seves tasques entorn al manteniment de la web.

S'estructura en tres punts bàsics segons el tipus de gestió que es vulgui realitzar, indicant en cada apartat els passos que s'han de seguir per portar a terme l'acció desitjada, acompanyant-los de les il·lustracions corresponents que facilitar-ne el seguiment i aconseguir l'objectiu desitjat.

Per tal de poder accedir a la gestió dels continguts caldrà que els usuaris encarregats del manteniment disposin d'un compte d'administració de la web.

Amb aquest compte i dirigint-se a l'adreça <http://opencms2.udl.es:8080/opencms/system/login/index.html> podran accedir a l'espai dedicat al lloc web de Griho dins de l'entorn de treball d'OpenCms, és a dir, la zona del sistema virtual de fitxers (VFS) on es troben les pàgines que conformen la web.


Els tres tipus de gestió per als quals s'ha creat aquesta guia són el manteniment de continguts, el manteniment de continguts estructurats i la creació de pàgines noves, detallats a continuació:

- **Manteniment dels continguts editables**

Tal com s'explica en el treball, els continguts editables són aquells que poden ser accedits i modificats des del navegador web mentre es visualitza la pàgina que els conté.

Així, el primer pas consistirà en fer clic sobre el nom de la pàgina en qüestió en el panell de detall de fitxers per poder-la visualitzar.

Un cop s'ha fet aquest pas, anem a enumerar el procediment per a poder editar els continguts:

1. Clicar a la icona d'edició directa  per accedir a l'editor FCKEditor.
2. Seleccionar l'idioma en el qual es volen actualitzar els continguts a través de la pestanya d'idioma.
3. Utilitzar les eines que proporciona aquest editor per posar la informació.
Les possibilitats que ofereix són molt semblants a les d'un processador de textos.
4. Desar els canvis.



Segons el botó escollit es realitzarà una acció diferent.

- La primera icona es correspon al botó de publicació directa, que desa del document, tanca l'editor i finalment publica la pàgina.
- La segona serveix per desar els canvis i sortir de l'editor. Cal para atenció a aquesta icona perquè la imatge sol crear confusions.
- La tercera icona és el botó per guardar, que és molt útil per desar els canvis periòdicament a mesura que s'està editant.
- La darrera icona és el botó per a tancar l'editor, que sortirà sense desar els canvis, tot i que sempre pregunta si realment es vol tancar abans de fer-ho, per evitar pèrdues indesitjades de la informació.

- **Manteniment dels continguts estructurats**

En el lloc web de GRIHO s'utilitzen tres tipus de continguts estructurats: les notícies, els esdeveniments i els membres.

Pels dos primers s'utilitzen els tipus predefinits que proporciona el sistema OpenCms, mentre que pels membres, s'aprofita l'estructura del tipus “news”, per tant, el procediment serà diferent.

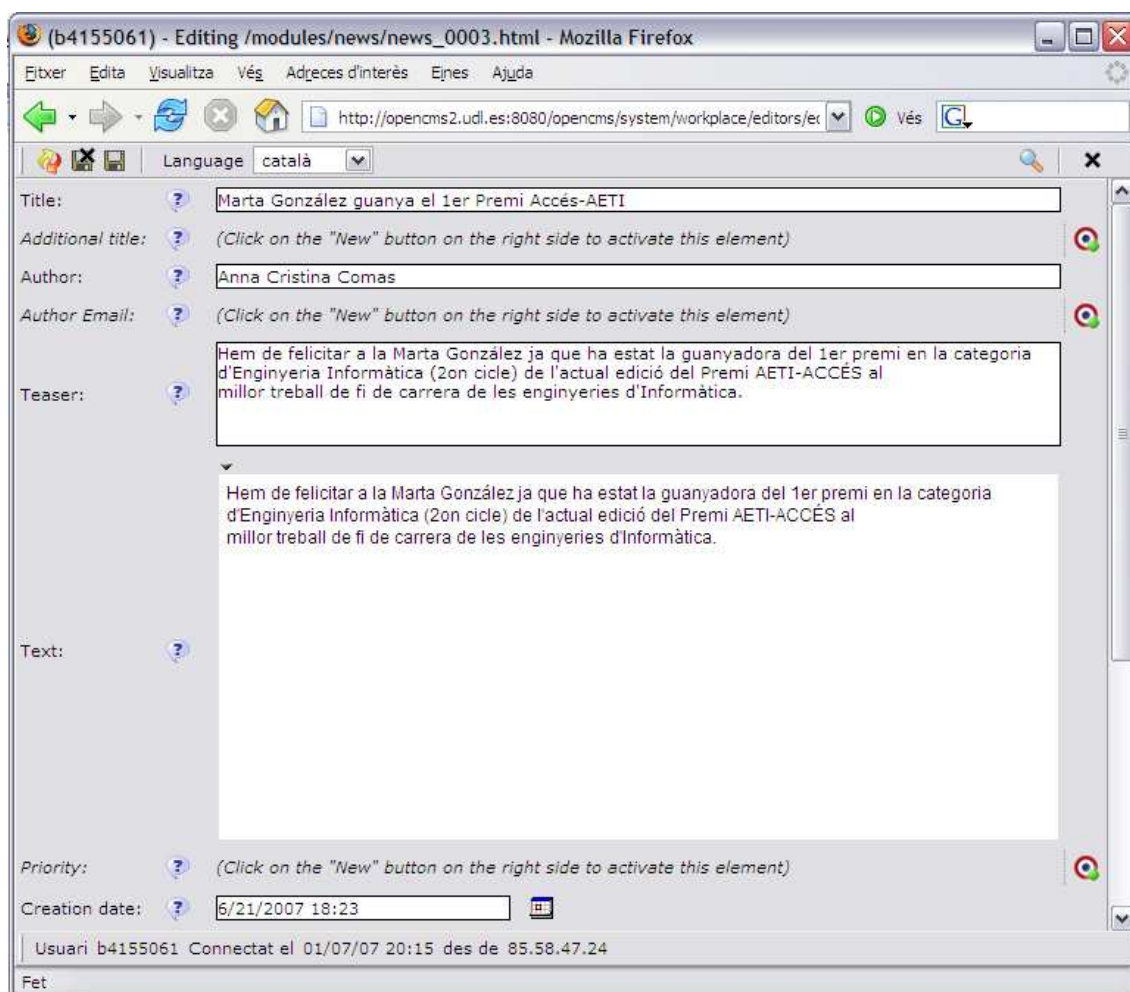
- **Notícies i Esdeveniments**

Per accedir a les funcions d'edició s'ha de visualitzar la pàgina que conté aquesta informació a través del navegador web, de la mateixa manera que en el cas dels continguts editables.

Visualitzant la pàgina, per a cada element hi haurà disponibles les següents icones:



- La primera icona es correspon al botó d'edició directa, amb el qual s'accedeix a l'editor.



Aquest editor és semblant al dels continguts editables per que fa a la selecció d'idiomes i en la manera com es desen els canvis. Per altra banda, per introduir la informació, caldrà omplir cadascun dels camps que ja vénen configurats.

Alguns d'aquests tenen una icona al costat que es desplega i funciona de la mateixa manera que les icones que s'estan descrivint ara dedicades al manteniment de continguts estructurats.



- La segona icona serveix per eliminar l'element.
- I la tercera icona permet crear un element nou del mateix tipus. Clicant-hi s'accedirà a l'editor d'aquest nou element.

Per tant, el procediment de gestió de notícies i esdeveniments serà el següent:

a) Per **editar** una notícia o un esdeveniment ja existent

1. Clicar sobre la primera icona per accedir a l'editor dels camps de contingut.
2. Seleccionar en la pestanya d'idioma l'idioma en el qual es volen actualitzar els continguts.
3. Fer els canvis pertinents.
4. Desar els canvis. (S'utilitzen els mateixos botons que en el cas dels

continguts editables)

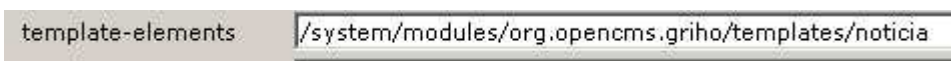
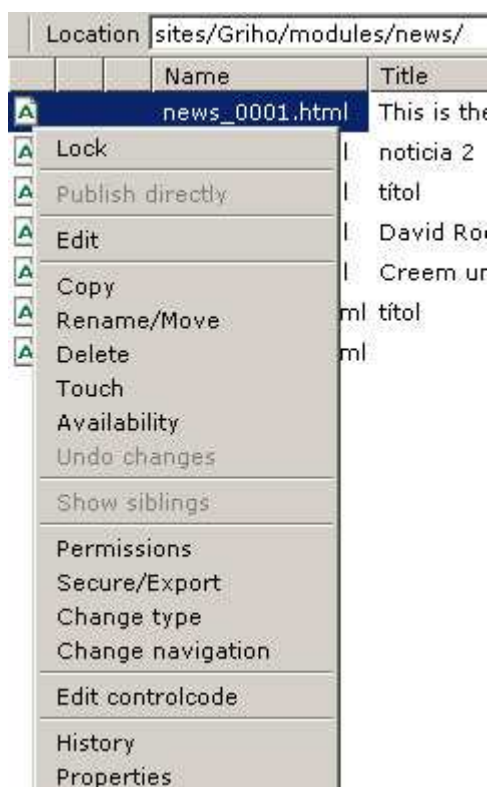


b) Per **eliminar** una notícia o un esdeveniment

1. Clicar sobre la segona icona.

c) Per **crear** una notícia o un esdeveniment nou

1. Clicar sobre la tercera icona.
2. Editar l'element seguint els passos 2, 3 i 4 d'edició (de l'apartat "a").
3. Cercar el fitxer que conté els continguts introduïts. Aquest fitxer es troba dins del directori arrel del lloc web, en la ruta `"/modules/news"` o `"/modules/events"`, segons el tipus creat.
4. Introduir la propietat `"template-elements"` a través del menú contextual del fitxer accedint a les propietats avançades. S'hi ha de posar la plantilla corresponent a "Notícia" o "Esdeveniment", segons el cas.



▪ Membres

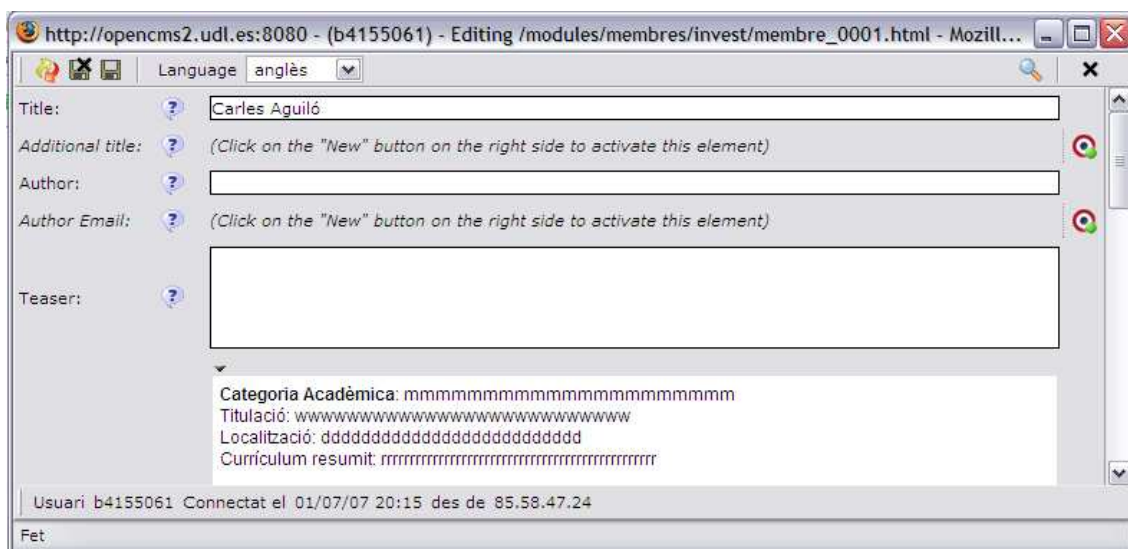
Per al manteniment dels membres s'ha d'accedir al directori “/modules/membres/[tipus_membre]” dins del sistema de fitxers del lloc web de GRIHO.

La gestió dels membres es fa de la manera següent:

a) Per **editar** un membre ja existent

1. Clicar sobre l'opció “Editar” del menú contextual del fitxer que conté la informació del membre.
2. Seleccionar en la pestanya d'idioma l'idioma en el qual es volen actualitzar els continguts.
3. Fer els canvis pertinents, tenint en compte les característiques especials:
 - En el camp “Title” s'hi posa el nom de la persona.

- En el camp “Text”, la descripció, posant en negreta els títols per poder obtenir l’aspecte desitjat en aplicar l’estil posteriorment.
- Es podran afegir arxius adjunts (un màxim de tres, si és possible, per evitar problemes de distribució de la pàgina).
- Es pot afegir un enllaç extern indicant la ruta cap a la pàgina personal, opcionalment.



4. Desar els canvis. (S'utilitzen els mateixos botons que en el cas dels continguts editables).

b) Per **eliminar** un membre

1. Clicar sobre l'opció de suprimir del menú contextual del fitxer.

c) Per **crear** un membre nou

1. Clicar sobre la icona de l'entorn de treball per crear recursos nous



, escollir el tipus “contingut estructurat”, i dins d'aquest, el tipus “news”.

2. Posar el nom del fitxer seguint el patró dels noms dels fitxers del directori de membres.
3. Editar el fitxer seguin els passos de l'apartat “a” d'aquesta secció.


4. Introduir la propietat “template-elements” a través del menú contextual del fitxer accedint a les propietats avançades. S’ha de posar la plantilla corresponent a “Membres”.

- **Creació de noves pàgines**

Caldrà tenir en compte les següents consideracions:

- La pàgina ha de pertànyer a un submenú nou (en cas contrari s’hauria de modificar el codi, ja que no estava previst segons l’estructura de la informació obtinguda de l’anàlisi de requisits: només es permeten 2 nivells).
- Dins del menú de Publicacions, degut a la càrrega d’informació externa dinàmicament serà necessari modificar el codi: els administradors de continguts no poden crear noves pàgines directament, ja que no funcionaran.

Per a la resta de pàgines, el procediment serà el que es descriu a continuació:

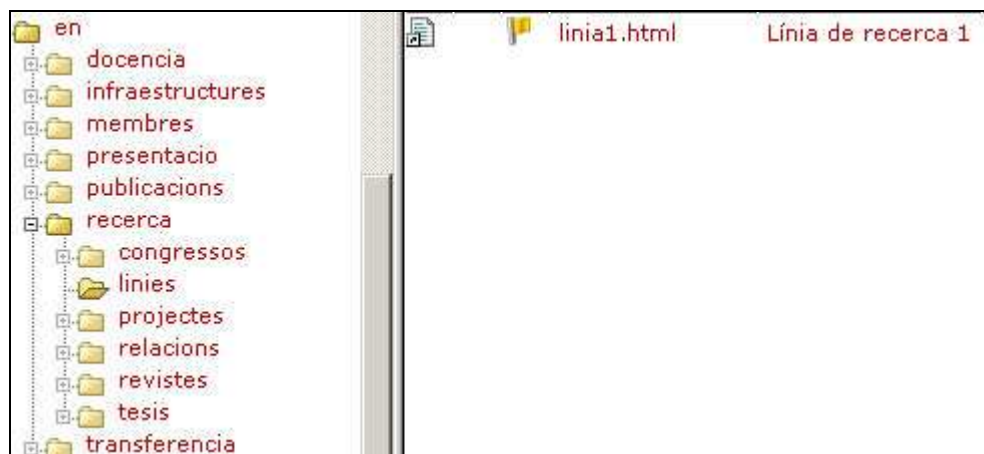
1. Dins de l’VFS, situar-se dins de la carpeta corresponent al punt de menú al qual ha de pertànyer la nova pàgina, en qualsevol dels directoris d’idioma del lloc web.
 - Si aquest menú té pàgina principal:
 1. Crear un directori nou per a posar la pàgina.
 2. Editar les propietats avançades del directori (s’hi accedeix amb el menú contextual):
 - “locale”: nom del directori d’idioma actual
 - “default-file”: nom del nou fitxer que es crearà ([nom].html)
 - “Title”: títol del submenú en l’idioma corresponent
 3. Crear el fitxer de tipus pàgina amb la icona  dins del directori.
 4. Editar les propietats de la pàgina:
 - “Title”: el títol

- “Template”: la plantilla que li volem aplicar (normalment el nom de la plantilla és representatiu del menú on s’ubica la pàgina)
- “locale”: nom del directori d’idioma actual (des de les propietats avançades)

| Property | Value | Used |
|-------------------|-------------------------------------|-------------------------------------|
| Template | Inici | <input checked="" type="checkbox"/> |
| Title | Inici | <input checked="" type="checkbox"/> |
| Keywords | | |
| Description | | |
| Add to navigation | <input checked="" type="checkbox"/> | |
| Navigation Text | | |
| Insert after | --- no change --- | |

locale ca ☒

5. Un cop creat el fitxer, repetir els passos 1 i 2 per cadascun dels directoris d’idioma.
6. Amb el menú contextual de la pàgina, crear una còpia germana als directoris creats al punt anterior i editar les propietats seguint el punt 4.



- Si no té pàgina principal, es posa la pàgina directament dins la carpeta de menú i es segueixen els passos 3, 4 i 6 del cas anterior.

2. Actualitzar el “linklist_submenu” corresponent:

- Crear un element nou de tipus “linklist” posant el títol i la ruta adequada cap a les pàgines creades en cadascun dels idiomes.



- Amb el menú contextual accedir a “Canviar navegació” i posar l’element al final o al lloc desitjat.
- Editar per cada idioma:
 1. El títol del submenu.
 2. La ruta cap a la pàgina creada en el camp “URL”.

(L'actualització del "linklist" també s'ha de tenir en compte quan s'elimini alguna pàgina, ja que si no es suprimeix el "linklist" associat apareixeran errors de pàgina no trobada).

3. Ja es pot editar la pàgina seguint el mateix procediment que per al manteniment de continguts editables (visualitzant-la en el navegador), ja que les possibilitats d'edició vindran donades per la plantilla que li hem aplicat.